R&S®Pulse Sequencer
Pulse Signal Generation Software
User Manual



1176.9512.02 ─ 03

Test & Measurement

User Manual

ROHDE&SCHWARZ

This document describes the following software options:

- R&S®SMW-K300/-K301
  1413.8805.02, 1413.9776.02
- R&S®SMBV-K300/K301
  1419.2744.02, 1419.2789.02
- R&S®SGT-K300/K301
  1419.7652.02, 1419.7700.02

© 2015 Rohde & Schwarz GmbH & Co. KG
Mühldorfstr. 15, 81671 München, Germany
Phone: +49 89 41 29 - 0
Fax: +49 89 41 29 12 164
E-mail: info@rohde-schwarz.com
Internet: www.rohde-schwarz.com
Subject to change – Data without tolerance limits is not binding.
R&S® is a registered trademark of Rohde & Schwarz GmbH & Co. KG.
Trade names are trademarks of the owners.

The following abbreviations are used throughout this manual: R&S®SMW200A is abbreviated as R&S SMW, R&S®SMBV100A is abbreviated as R&S SMBV, R&S®Pulse Sequencer is abbreviated as R&S Pulse Sequencer, R&S®WinIQSIM2™ is abbreviated as R&S WinIQSIM2.

Contents

1 Preface 9

1.1 About this Manual 9

1.2 Documentation Overview 10

1.3 Conventions Used in the Documentation 11

1.3.1 Typographical Conventions 11

1.3.2 Conventions for Procedure Descriptions 11

1.3.3 Notes on Screenshots 12

2 Welcome to the R&S Pulse Sequencer 13

2.1 Accessing the Pulse Sequencer 14

3 Getting Started 15

3.1 Required Options 15

3.2 Installing the Software 16

3.3 Starting the R&S Pulse Sequencer for the First Time 18

3.4 Understanding the Displayed Information 19

3.5 Means of Users Interaction 22

3.6 Trying Out the Software 25

3.6.1 Completing the Scenario that was Automatically Created upon Start Up 26

3.6.2 Generating an ARB Waveform File 27

3.6.3 Verifying the Generated Signal in the Waveform View 28

3.6.4 Launching the Built-In Wizard 29

3.6.5 Using the Wizard to Create a Complex Scenario 30

3.6.6 Transferring the Waveform to and Playing it with a Connected Instrument 33

3.6.7 Saving and Recalling Settings 34

3.6.8 Advanced Features and Examples 34

3.7 Customizing the Software 35

3.7.1 Customizing Your Workspace 35

3.7.2 Changing Colors and Default Configuration 36

3.8 Getting Information and Help 38

4 About the R&S Pulse Sequencer 39

5 Organizing the Project Data in Repositories 42

5.1 Repository Settings..... 42
5.2 Repository Manager Settings..... 46
5.3 How to Manage the Project Data..... 48
6 Selecting a Suitable Scenario and Creating Scenarios.....53
6.1 Overview of the Available Scenarios and Their Complexity.....53
6.2 Scenario Settings.....54
6.2.1 Common Scenario Settings..... 55
6.2.2 Scenario Specific Settings..... 60
6.3 How to Select and Create a Test Scenario..... 63
7 Creating a Pulse Library.....66
7.1 Basics on Pulse Signals and Pulse Generation.....66
7.2 Pulse Settings..... 68
7.2.1 Pulse Name and Comment.....69
7.2.2 Pulse Timing Settings..... 70
7.2.3 Pulse Level Settings.....73
7.2.4 Pulse Envelope Settings..... 76
7.2.5 Modulation on Pulse (MOP) Settings.....79
7.2.6 Built-In Modulation Types and their Settings.....82
7.2.6.1 AM and ASK Modulation.....83
7.2.6.2 FM and FSK Modulation..... 85
7.2.6.3 Chirp Modulation.....87
7.2.6.4 Phase Modulation..... 89
7.2.6.5 QPSK Modulation.....92
7.2.6.6 Noise.....93
7.2.6.7 Plugins..... 93
7.2.6.8 Data Source, Coding, Filter.....95
7.2.7 Envelope and Modulation Graphs.....96
7.3 How to Create a New Pulse and Adjust Its Settings.....98
8 Building Pulse Sequences..... 106
8.1 About the Sequencing Principles.....106
8.2 Sequence Settings.....108
8.2.1 Sequence Description Settings..... 109

8.2.2 Pulse Repetition Settings....112
8.2.3 Loop Settings....113
8.2.4 Overlay Settings....114
8.2.5 Fillers Settings....114
8.2.6 Lists with Multiple Sequences....116
8.3 How to Create Sequences and Use the Control Elements....117

9 Defining and Enabling Inter-Pulse Modulation Effects....127
9.1 IPM Profiles Settings....130
9.1.1 Common IPM Settings....130
9.1.2 IPM Profiles Settings....132
9.1.3 Edit List Settings....140
9.1.4 Configure Inter-Pulse Modulation (IPM) Settings....142
9.2 How to Create IPM Profiles and Use Them to Vary Pulse Parameters....144

10 Defining Antenna Patterns and Antenna Scans....154
10.1 Overview of the Main Antenna Parameters....154
10.2 Antenna Pattern Settings....155
10.3 Antenna Scans Settings....164
10.4 How to Create a Library with Antenna Patterns and Scans....171

11 Emulating Emitters....179
11.1 Emitter Settings....180
11.2 3D Emitter Preview Settings....184
11.3 Lists with Multiple Emitters....185
11.4 How to Create and Configure Emitters....186

12 Working with Waveforms and Generating Interfering Signals....192
12.1 Waveform Settings....193
12.2 How to Create a Waveform Scenario and Work with Waveforms....195

13 Creating Complex 2D Scenarios with Receivers and Interferers..201
13.1 Related Settings....202
13.1.1 2D Map Settings....202
13.1.2 Receiver Settings....204
13.1.3 3D Scan Pair View Settings....205
13.1.4 Lists with multiple emitters and interferers....206

13.2 How to Create Scenarios with Emitters, Interferers and a Receiver..... 209

14 Creating Generator Profiles and Configuring the Instruments..... 216

14.1 Instrument Configuration Settings..... 217

14.2 Generator Profile Settings..... 219

14.3 How to Create Generator Profiles and Configure the Connected Instruments...221

15 Defining the Data Source..... 226

15.1 Data Sources Settings..... 228

15.2 How to Configure the Bit Stream Used by the MOP..... 230

16 Defining Complex Modulation Schemes and IPM Profiles.....232

17 Defining and Enabling Marker Signals.....236

18 Visualizing and Analyzing Waveforms.....240

18.1 Waveform and Data View Settings..... 240

18.2 Waveform Reference Level Settings..... 243

18.3 How to Analyze the Content of Waveform Files and Files with I/Q Data..... 245

19 Playing the Generated Waveform Files.....247

19.1 Waveform Generation Settings.....247

19.2 Signal to Generator Mapping Settings.....250

19.3 Signal Generator Remote Control Settings.....252

19.4 Playing Waveforms with the Signal Generator..... 253

19.4.1 How to Create, Transfer and Play the Waveforms of Simple Scenarios.....253

19.4.2 How to Play the Waveforms of Emitter-Based 2D Scenarios..... 255

20 Creating Reports and Documenting Measurement Results..... 260

20.1 Reporting Settings.....261

20.2 How to Create Test Reports.....263

21 Automation of R&S Pulse Sequencer..... 266

21.1 How to Configure and Enable Remote Control of R&S Pulse Sequencer..... 267

22 Remote-Control Commands..... 269

22.1 Conventions used in SCPI Command Descriptions..... 269

22.2 Programming Examples.....270

22.3 Common Commands.....270

22.4 Commands with Similar Syntax....271
22.5 Antenna Pattern Commands....283
22.6 Antenna Scan Commands....291
22.7 Data Source Commands....299
22.8 Emitter to Generator Mapping Commands....301
22.9 Emitter Commands....304
22.10 Generator Profiles and Instruments Commands....307
22.11 Inter Pulse Modulation Commands....310
22.12 Marker Commands....322
22.13 Plugin and Reporting Commands....326
22.14 Pulse Commands....331
22.15 Repository Commands....350
22.16 Scenario Commands....357
22.17 Sequence Commands....372
22.18 Status Commands....381
22.19 System Commands....382
22.20 Waveform Commands....384
22.21 Waveform Viewer Commands....385

23 Querying Error Messages and Troubleshooting....388

A Annex....390
A.1 Supported File Types and File Formats....390
A.1.1 File Format of the Reporting Template....390
A.1.2 Antenna Pattern File Formats....393
A.2 Formula Syntax....398
A.3 Plugin Programming API....400
A.3.1 Common Functions....401
A.3.2 MOP Functions....404
A.3.3 IPM Functions....409
A.4 Common Coding Algorithms....410
A.5 Supported Filter and Impact of the Filter Parameters....410
A.6 References....412

Glossary: Terms and Abbreviations....413

List of Commands.....416

Index.....428

1 Preface

1.1 About this Manual

This User Manual provides all the information **specific to the R&S Pulse Sequencer** and describes the software and all its options.

This description assumes that all available options are installed. Required options are stated at the beginning of each section or in the parameter description, if one particular parameter required additional option. If your software does not support all options, consider the corresponding parts as not relevant.

The main focus in this manual is on the provided settings and the tasks required to generate a signal. The following topics are included:

- **Welcome to the R&S Pulse Sequencer options R&S SMW/SMBV/SGT-K300/-K301**
  Introduction to and getting familiar with the option
- **Getting Started**
  Information needed to install and start working with the software, as well as basic operation and typical signal generation examples
- **About the R&S Pulse Sequencer**
  Background information on basic terms and principles in the context of the signal generation
- **R&S Pulse Sequencer Configuration and Settings**, logically divided in several topics
  A concise description of all functions and settings available to configure signal generation with their corresponding remote control commands
- **How to Generate a Signal with the R&S Pulse Sequencer Options**
  The basic procedure to perform signal generation tasks and step-by-step instructions for more complex tasks or alternative methods.
  Detailed examples guide you through typical signal generation scenarios and allow you to try out the application immediately
- **Typical Application Examples**
  Example signal generation scenarios in which the software is frequently used.
- **Remote Control Commands**
  Remote commands required to configure and perform signal generation in a remote environment, sorted by tasks
  Programming examples demonstrate the use of many commands and can usually be executed directly for test purposes
  (Commands required to set up the vector signal generator or to perform common tasks on the instrument are provided in the User Manual of the base unit)
- **Annex**
  Reference material, such as extensive lists, tables or overviews
- **Glossary**
  Alphabetical list of often used terms and abbreviations

- **List of remote commands**
  Alphabetical list of all remote commands described in the manual
- **Index**

1.2 Documentation Overview

The user documentation for the R&S Pulse Sequencer consists of the following parts:

- Online Help system included in the software
- User Manual
- Release Notes
- Web Help
- Application Notes

Online Help

The Online Help is embedded in the software. It offers quick, context-sensitive access to the complete information needed for operation and programming. The online help contains help on operating the R&S Pulse Sequencer and all available options.

User Manual

The user manual describes the R&S Pulse Sequencer and all its options. It provides basic information on operating the R&S Pulse Sequencer in general. In this manual, all functions are described in detail together with examples on some specific tasks. Furthermore, it provides a complete description of the remote control commands with programming examples.

The user manual is available for download from the R&S website, on the product page of the signal generator. For example at http://www.rohde-schwarz.com/product/SMW200A.html.

Release Notes

The release notes describe the installation of the software, new and modified functions, eliminated problems, and last minute changes to the documentation. The corresponding software version is indicated on the title page of the release notes.

The most recent release notes are available for download from the R&S website, on the signal generator's product page. For example at http://www.rohde-schwarz.com/product/SMW200A.html > Downloads > Firmware.

Web Help

The web help provides online access to the complete information on operating the R&S Pulse Sequencer and all available options, without downloading. The content of the web help corresponds to the user manual for the latest product version.

The web help is available from the signal generator's product page. For example at http://www.rohde-schwarz.com/product/SMW200A.html > Downloads > Web Help.

Application Notes

Application Notes, Application Cards, White Papers and Educational Notes are further publications that provide more comprehensive description of specific test configurations as well as the background information on special technical topics.

Consider the following sample of further useful information:

- Application Note 1MA211 "Coexistence Test of LTE and Radar Systems"
- Application Note 1MA256 "Real-time Radar target Generation"
- Application Card "Simplify pulse and emitter generation for radar testing"
- Application Card "Analyze your radar signals with wideband power sensors"
- Application Card "Flexible target generation for radar receiver testing"
- Application Card "Simplified analysis of frequency agile radars"

The latest versions of application notes and program files are available for download from the R&S website, at the http://www.rohde-schwarz.com/appnotes.

1.3 Conventions Used in the Documentation

1.3.1 Typographical Conventions

The following text markers are used throughout this documentation:

Convention	Description
"Graphical user interface elements"	All names of graphical user interface elements on the screen, such as dialog boxes, menus, options, buttons, and softkeys are enclosed by quotation marks.
KEYS	Key names are written in capital letters.
`File names, commands, program code`	File names, commands, coding samples and screen output are distinguished by their font.
Input	Input to be entered by the user is displayed in italics.
Links	Links that you can click are displayed in blue font.
"References"	References to other parts of the documentation are enclosed by quotation marks.

1.3.2 Conventions for Procedure Descriptions

When describing how to operate the software, several alternative methods may be available to perform the same task. In this case, the procedure using the menu bar is described. Many elements that can be accessed form menus can also be accessed by clicking an icon or using the context menu. Alternative procedures are only described if they deviate from the standard operating procedures.

The term "select" may refer to any of the described methods.

1.3.3 Notes on Screenshots

When describing the functions of the product, we use sample screenshots. These screenshots are meant to illustrate as much as possible of the provided functions and possible interdependencies between parameters. The shown values may not represent realistic test situations.

The screenshots usually show a fully equipped product, that is: with all options installed. Thus, some functions shown in the screenshots may not be available in your particular product configuration.

2 Welcome to the R&S Pulse Sequencer

The R&S Pulse Sequencer is a software that generates complex pulse signals by using predefined, configurable test scenarios with different complexity. You can simulate the signals of different emitter and receiver configurations, including antennas and scan types. The signal can be played by the instruments listed below, for example to perform radar receiver tests.

- R&S SMW
- R&S SMBV
- R&S SGT

The R&S Pulse Sequencer software is a stand alone, PC based application that creates waveform files. This software is available for free download on the R&S website.

Pulse Sequencer features:

- ARB based signal generation
- Multi segment waveform sequencing
- Pulse shape definition with rise and fall time, droop, ripple, overshoot
- Modulation on pulse with all major formats like chirps, Barker Codes, Polyphase Codes, PSKs, AM, FM
- Single pulse, pulse train generation with repetition count per pulse
- Inter pulse modulation of amplitude, phase, frequency, etc. values from pulse to pulse
- Internal and external data sources for modulation
- Import of waveform files for sequencing with repetition count

Advanced Pulse Sequencer key features:

- ARB based signal generation and multi segment waveform sequencing
- Single pulse and pulse train generation with repetition count per pulse
- Powerful sequencing tool with loops, nested loops, subsequences and overlays
- Antenna diagram definition and antenna scan definition
- Antenna diagrams like pencil beams, cosecans beams, Gaussian diagrams, user defined antenna diagrams, phased array antenna diagrams
- Antenna scan types like helical scans, circular scans, conical scans
- Emitter definition by waveforms, antenna diagram, antenna scan, attitude information, EIRP and carrier frequency
- Receiver definition by antenna diagram, antenna scan and attitude information
- Calculation of signal under consideration of one-way free space propagation according to emitter and receiver location on the 2D map
- Import of R&S®WinIQSIM2™ or customer waveforms for interference generation on the 2D map

This user manual contains a description of the functionality that the software provides, including remote control operation.

Functions of the signal generator are not described here; they are described in the user manual of the corresponding base unit. The latest version is available for download at the instrument's page, e.g. at R&S SMW200A product homepage.

Installation

You can find detailed installation instructions in the delivery of the option or in chapter 3.2, "Installing the Software", on page 16.

2.1 Accessing the Pulse Sequencer

To open the application

► On your PC, select "Start > All Programs > R&S Pulse Sequencer > R&S Pulse Sequencer".

The software opens and per default displays the last opened workspace.

3 Getting Started

This section describes the basic steps to be taken when starting up the R&S Pulse Sequencer for the first time.

- Required Options 15
- Installing the Software 16
- Starting the R&S Pulse Sequencer for the First Time 18
- Understanding the Displayed Information 19
- Means of Users Interaction 22
- Trying Out the Software 25
- Customizing the Software 35
- Getting Information and Help 38

3.1 Required Options

The R&S Pulse Sequencer software is a stand alone, PC based application that creates waveform files. This software is available for free download on the R&S website. Waveforms generated with the software can be played by a subset of instruments, that must fulfill the minimum requirements listed below.

R&S SMW200A

R&S SMW200A base unit equipped with:

- option Baseband Generator (R&S SMW-B10)
- option Frequency (R&S SMW-B1xx/-B2xx)
- option Baseband main module, one I/Q path to RF (R&S SMW-B13)
- option Pulse Sequencer (R&S SMW-K300) per signal path

Additional non mandatory options:

- option Advanced Features for Pulse Sequencer (R&S SMW-K301) per signal path
- further options depending on the particular application
  e.g. if waveform files generated with R&S®WinIQSIM2™ will be used within the sequences, the corresponding R&S SMW-K2xx options

R&S SMBV100A

R&S SMBV100A base unit equipped with:

- option Baseband Generator (R&S SMBV-B51)
- option Memory Extension for ARB (R&S SMBV-K511)
- option Frequency (R&S SMBV-B1xx)
- option Pulse Sequencer (R&S SMBV-K300)

Additional non mandatory options:

- option RF Bandwidth Extension (R&S SMBV-K521/-K522)
- option Advanced Features for Pulse Sequencer (R&S SMBV-K301)

- further options depending on the particular application
  e.g. if waveform files generated with R&S®WinIQSIM2™ will be used within the sequences, the corresponding R&S SMBV-K2xx options

R&S SGT100A

R&S SGT100A base unit equipped with:

- option Frequency (R&S SGT-KB106)
- option Baseband Generator (R&S SGT-K510)
- option Pulse Sequencer (R&S SGT-K300)

Additional non mandatory options:

- option Memory Extension for ARB (R&S SGT-K51x)
- option Advanced Features for Pulse Sequencer (R&S SGT-K301)
- further options depending on the particular application
  e.g. if waveform files generated with R&S®WinIQSIM2™ will be used within the sequences, the corresponding R&S SGT-K2xx options

3.2 Installing the Software

Software updates as well as the Release Notes describing the improvements and modifications are available for download at the product homepage of the signal generator, e.g. R&S SMW.

Software and Hardware Requirements

The following minimum requirements have to be met:

- **Software**
  - Microsoft®Window 7 Professional 32 Bit (SP1) operating system
  - Rohde&Schwarz VISA IO Libraries for Instrument Control or other VISA runtime library, e.g. National Instruments VISA 4.0. or higher.
- **Hardware**

Table 3-1: Hardware requirements

	Minimum requirements	**Recommended hardware**
AMD or Intel CPU	Dual Core, 2 GHz	Intel Quad-Core i7, Quad-Core Xeon, AMD FX series
RAM	≥ 2 GB	16 GB
Video	NVIDIA Quadro 128 MB or ATI Radeon	NVIDIA Quadro 128 MB or ATI Radeon
Video resolution	≥ 1280 x 1024 pixels	≥ 1920 x 1200 pixels
Rendering	OpenGL, shader model 3	OpenGL, shader model 3
Network	LAN 1 GB/s	LAN 1 GB/s

To install the software

Download the R&S Pulse Sequencer software form the R&S website.

The R&S Pulse Sequencer software consists of the file
`PS-Install <major>.<minor>.<build>.<release>.msi`.

1. Start the `PS-Install <major>.<minor>.<build>.<release>.msi` file.

   The file name follows the naming conventions:

   - `<major>.<minor>` is the software version
   - `<build>` is the build number.
     This is the day elapsed since 1 Jan. 2000
   - `<release>` indicates the release on the build date.
     This is 1/10 of the number of seconds elapsed since midnight.

2. Depending on your user rights, select one of the following:

   a) Installation for all users (default)

      - Requires administrator rights
      - Is a per-machine installation
      - Project data and program data are common to all users
      - Users keep their Individual settings

   b) Installation for current user only

      - Does not require administrator rights
      - Is a per-user installation
      - Installation in `HomePath`

3. Choose the setup type.

4. Follow the installation instructions.

   The installation is competed. Per default, a shortcut icon is created on the desktop.

The software creates the folder structure listed in table 3-2.

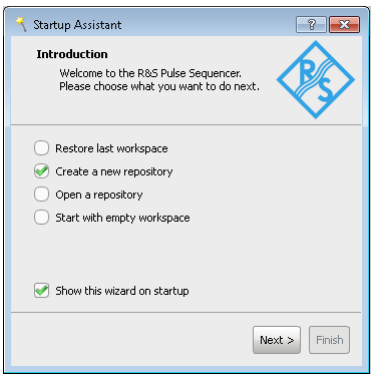
Table 3-2: Default file location (software installation for all users)

File type	File location
Program data, SDK files	`%PROGRAMFILES(X86)%\Rohde-Schwarz\Pulse Sequencer` (`InstallPath`)
Project data (Repositories)	`%PUBLIC%\Public Documents\Rohde-Schwarz\Pulse Sequencer` (`RepositoryPath`)
Report files	`%HOMEPATH%` (`HomePath`)
User settings (Workspace, etc.)	`%HOMEPATH%\AppData\Roaming\Rohde-Schwarz\Pulse Sequencer`

3.3 Starting the R&S Pulse Sequencer for the First Time

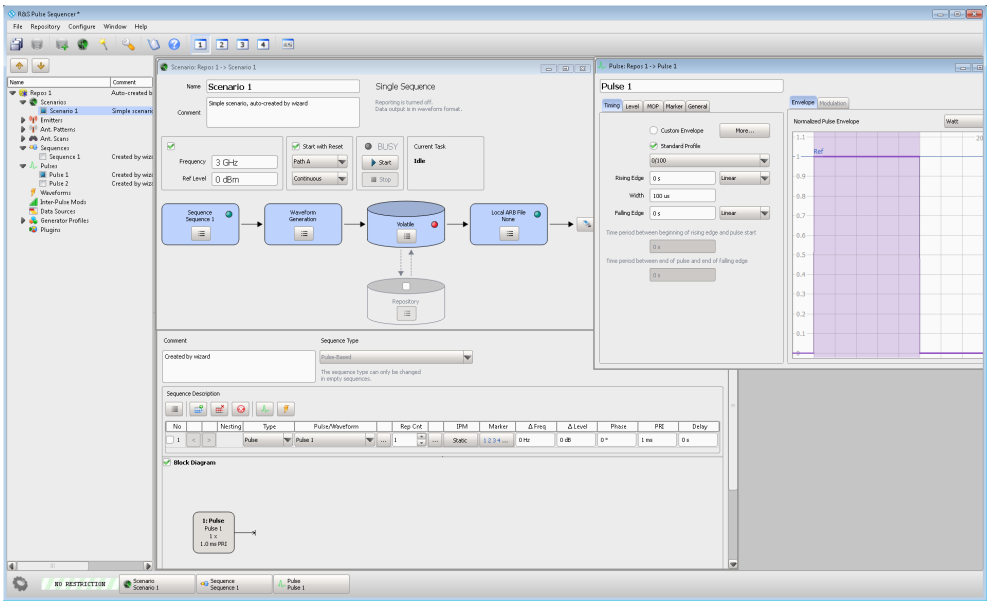
1. On your PC,
   a) select "Start > All Programs > R&S Pulse Sequencer > R&S Pulse Sequencer"
   or
   b) double click on the shortcut icon on the desktop.

   The "Startup Assistant" opens.



2. Select "Create a new Repository" and select "Next".
3. Use the default settings and follow the instructions.

   A new repository ("Repos 1") is created that contains a simple scenario ("Scenario 1").
   Your workspace should resemble the one shown on the following figure.



See also:

- figure 3-1 (in chapter 3.4, "Understanding the Displayed Information", on page 19)

- About the R&S Pulse Sequencer, section "Introduction to the Software Concept" on page 39
- chapter 3.6, "Trying Out the Software", on page 25

3.4 Understanding the Displayed Information

The following figure shows an example test scenario during software operation. All different information areas are labeled. They are explained in more detail in the following sections.

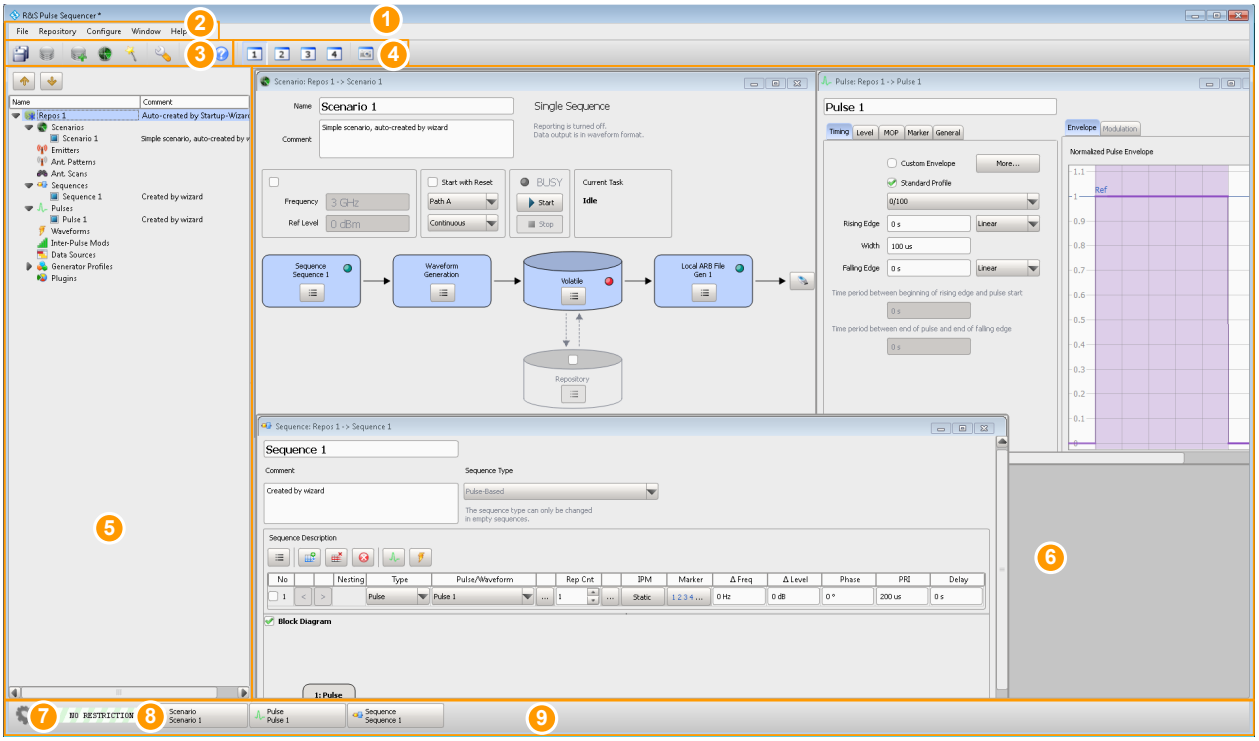


Fig. 3-1: R&S Pulse Sequencer default workspace: understanding the displayed information

1 = Title bar
2 = Menu bar with standard functions, e.g. save/open file etc.
3 = Toolbar
4 = Screens for individual scenarios
5 = Project tree (repository tree)
6 = Work area
5+6 = Workspace
7, 8, 9 = Task bar with indication on enabled remote control, information on the required highest security level (if enabled) and access to active dialogs

The user interface of the R&S Pulse Sequencer offers an intuitive operating concept.All menus and dialogs use known elements, e.g., selection lists, check boxes, and entry fields.

Menu bar

Table 3-3: Content of the menu bar

File	Repository	Configure	Window	Help
• New Repository • Load and Manage Repository • Import Repository Archive • Save Repository • Clear Workspace • Open Workspace • Save Workspace • Save Workspace As • Exit	quick access to currently loaded repository	• Instruments • Remote Control • Settings • Colors	• Console • Message Log • Cascade • Minimize All • Close All • Screen 1 .. Screen 4	• About • User Manual • Wizard

Toolbar

Standard functions can be performed via the icons in the toolbar at the top of the screen.



Fig. 3-2: Standard functions in the toolbar

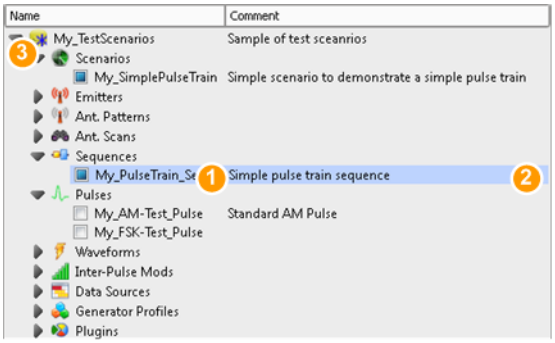
1 = Save as: stores current workspace configuration
2 = Access the "Manage Repository" dialog for loading, discarding or exporting repositories
3 = Creates a new repository
4 = Creates a new scenario
5 = Starts a wizard
6 = Access the "Program Settings" dialog
7 = Opens the message log window
8 = Help: displays context-sensitive help topic for currently selected element
9 = Screen: switches between the virtual screens
10 = Hardcopy: copies current dialog to the clipboard

Virtual screens

Using the R&S Pulse Sequencer you can handle several scenarios with even comprehensive settings at the same time. You can arrange logically the dialogs across multiple screens, so that you easily switch between them. You can for example assign the subset of antenna dialogs to one screen and the pulse and modulation dialogs to another.

Repository tree

The repository tree shows the content of all currently loaded repositories. Repository elements are indicated with the assigned icons, their names and the first comment line.

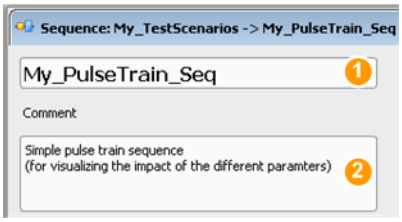


1 = Name, see "Name and comment" on page 21
2 = First comment line
3 = Indicates unsaved changes in the repository

Name and comment

Each repository element is identified with its unique name and optional description entered in the comment field. The description may contain several lines; new line are added by pressing SHIFT+ENTER.

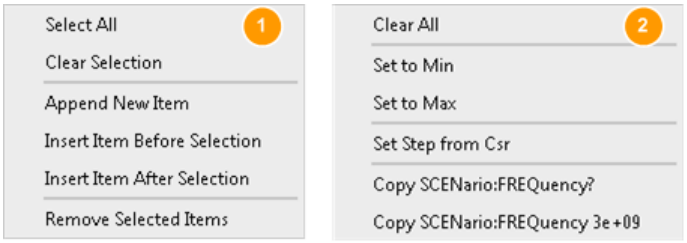
See also "Dialog names" on page 22.



1 = An unique name to indicate the element in the repository
2 = Description; the first line is displayed in the repository tree

Context-sensitive menus

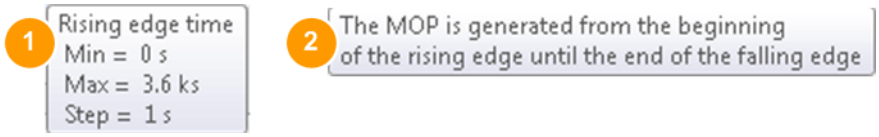
All user interface controls provide a context-sensitive menu. Context-sensitive menus are accessed with a right mouse click on the control.



1 = Context menu of an icon
2 = Context menu of a parameter

Tooltips

Tooltips indicate the possible value range of a parameter or list overview information on current settings.

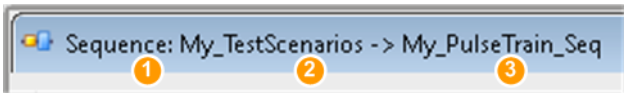

1
Rising edge time
Min = 0 s
Max = 3.6 ks
Step = 1 s
2
The MOP is generated from the beginning
of the rising edge until the end of the falling edge

1 = Tooltip of a setting parameter, indicating the allowed value range
2 = Tooltip with detailed information on a setting

Dialog names

The name of a dialog indicates its content or the functionality this dialog covers. The used naming convention helps you to recognize which data base element you are configuring, to which scenario does it belong and what is its current name.

"<Repository Element>: <Scenario Name> -> <Function Name>"

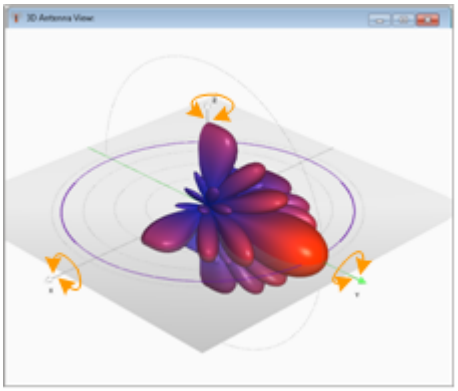


1 = Repository element
2 = Scenario name
3 = Function name

Interactive 3D displays

Several dialogs display the configured settings as 2D or as interactive 3D models or full 3D antenna radiation patterns.

Turning the mouse wheel zooms in and out on these dialogs. Moving the mouse while holding the left mouse key rotates the 3D view around its origin.



3.5 Means of Users Interaction

This chapter provides an overview on how to work with the R&S Pulse Sequencer and describes the manual operation of the software and also the alternative ways of operation.

There are two ways to operate the R&S Pulse Sequencer:

- Manual operation:
  Run the software on your PC and use the mouse and/or keyboard.

- Remote control:
  Create programs to automate settings, tests and measurements. The software is controlled by a program, running on the same or on another computer.
  This way of operation is described chapter 21, "Automation of R&S Pulse Sequencer", on page 266.

Basic operation concept

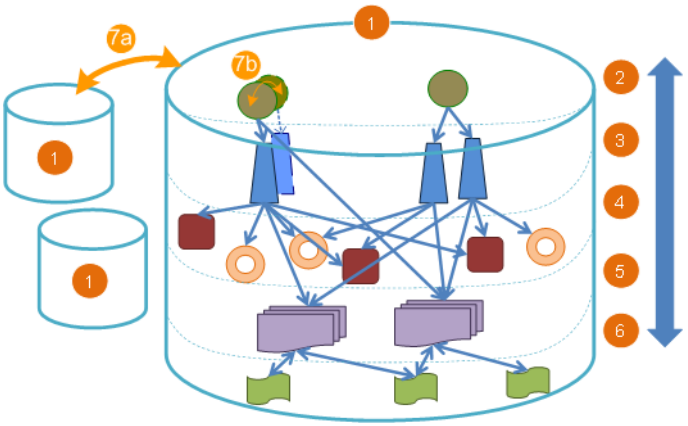
This section lists settings and functions that are common to the dialogs and are performed in the same manner throughout the software. You can use them alternatively or complementary.

As described in "Introduction to the Software Concept" on page 39, the R&S Pulse Sequencer software is based on a relational database that organizes database elements within repositories. Database elements are indicated by their names; within one repository, element names must be unique (see "Name and comment" on page 21).

Creating elements

Once created, database elements can be used by other elements within the same repository. Elements can be created in both the bottom-up and the top-down direction, or in a combination of these two methods:

- *Bottom-up:* you can create your pulse and waveform library first, then build sequences, create scenarios and use the created sequences or assign them to emitters
- *Top-down:* create your scenario, create new emitter, define the antenna pattern/ scan, define the used sequence and define the pulses and waveforms



1 = Repository, one or more
2 = Scenario level; scenarios describe one or more emitters and one ore more sequences
3 = Emitter level; emitters are described among others with antenna patterns and antenna scans
4 = Antenna patterns and antenna scans
5 = Sequence level; sequences are build of one or more pulses or waveforms
6 = Pulse and waveform level
7a, 7b = Creating elements at any level with clone and copy

For information on the naming conventions and description of the database elements, see "Introduction to the Software Concept" on page 39.

Handling database elements

While creating database elements, consider also the following possibilities:

- Change
  Files describing elements are stored once but may be used several time. When one file is changed, its parent elements are also updated.
- Clone
  Elements can be cloned to create a deep copy of the database element
- Delete
  Elements can be deleted if they are not referenced by other elements
- Copy
  If two repositories are opened in the project tree, a drag&drop operation creates a copy of the selected element and all referenced elements

Standard function in the context menus

Each dialog with settings belonging to a „higher-level“ element provides standard functions to:

- assign existing elements
- create new elements
- edit the existing assigned element

You can access these function from the context sensitive menus of the icons listed in table 3-4.

Table 3-4: Overview of the used icons and their functions

Icon	Name	Function/Description
	<Function> Menu (the name changes depending on the situation)	Smart menu with several options, depending on the current element and situation: • New (Item) • Append item • Prepend item • Insert Item • Insert Item Before/After • Delete Item • Clear (Selection) • Edit • Select • Rename
	More	Select Edit New
	Pulse, Waveform	Creates new repository element
	Antenna Pattern, Antenna Scan, Sequence	Accesses a dialog with detailed settings
	Emitter, interferer, receiver	

Icon	Name	Function/Description
	3D, 2D diagram	
	Append item, Remove item, Delete	Appends or removes item, deletes all items
	Left/Right , Up/Down	Reorders items
	Select	Selects a table row or an item for editing
	Enable/Disable	Activates or deactivates settings
	Hardcopy, Print	Copies current screen or dialog to the clipboard; Prints out current diagram

Standard functions for elements handling

Supported are the following standard direct interaction functions:

- Double-click
  Opens the selected element for editing
- Clone
  Creates a copy of the selected element
- Drag&Drop
  A drag&drop operation:
  – creates a copy of the selected element, together with all referenced elements
  – assigns an element to a 2D view
- Mouse wheel
  Turning the mouse wheel zooms in and out

3.6 Trying Out the Software

This chapter introduces the most important functions and often used settings of the R&S Pulse Sequencer step by step. The complete description of all functions and their usage is provided in the corresponding main chapters of this user manual.

Prerequisite

The software is running and started up as described in chapter 3.3, "Starting the R&S Pulse Sequencer for the First Time", on page 18.

For the first signal generation tasks, you use the *on-demand Wizard* to generate an ARB signal, so you do not need any external instruments. More complex signal generation tasks, however, may require an external signal generator equipped with the options listed in chapter 3.1, "Required Options", on page 15.

The software is manually operated. Try out the following:

- Completing the Scenario that was Automatically Created upon Start Up.....26
- Generating an ARB Waveform File.....27
- Verifying the Generated Signal in the Waveform View.....28
- Launching the Built-In Wizard.....29
- Using the Wizard to Create a Complex Scenario.....30
- Transferring the Waveform to and Playing it with a Connected Instrument.....33
- Saving and Recalling Settings.....34
- Advanced Features and Examples.....34

3.6.1 Completing the Scenario that was Automatically Created upon Start Up

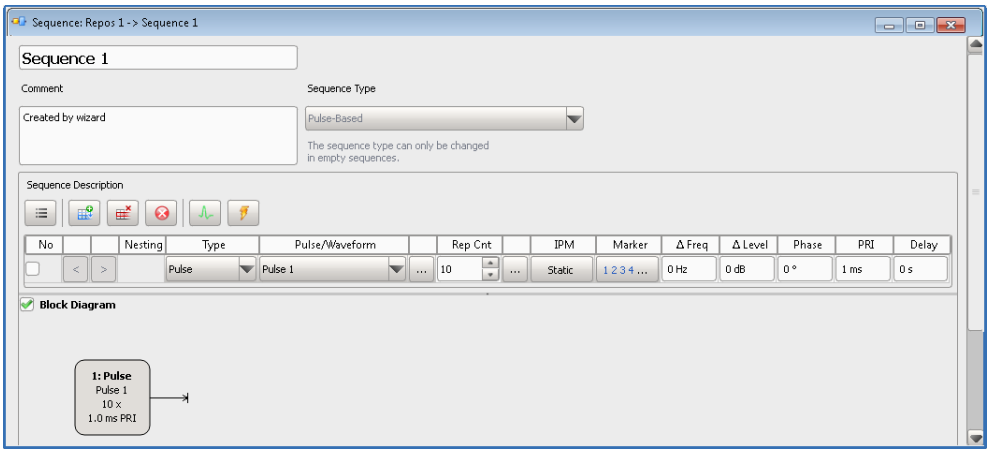
We assume that you have started the software as described in chapter 3.3, "Starting the R&S Pulse Sequencer for the First Time", on page 18. Your workspace should resemble the one shown on figure 3-1 (see section chapter 3.4, "Understanding the Displayed Information", on page 19).

Your repository contains:

- "Pulse 1": an unmodulated pulse with pulse width of 100 us and rise and fall times set to zero
- "Sequence 1": a sequence containing one single pulse, "Pulse 1". This pulse is processed once ("Rep. Count = 1") and has a pulse repetition interval "PRI = 3 ms".

To create a sequence with 10 pulses, each having a PRI of 1 ms

1. In the repository tree, select "Scenario > Sequence 1".
2. In the "Sequence Description" table, select:
   a) "Rep. Count = 10"
   b) "PRI = 1 ms"



For information on the provided settings, see:

- chapter 8.2, "Sequence Settings", on page 108
- chapter 7.2.2, "Pulse Timing Settings", on page 70

To assign the sequence to the scenario

1. In the repository tree, select "Scenario > Scenario 1".

   The "Scenario" dialog opens and displays a block diagram of the signal flow.

2. In the block diagram, select "Sequence" and select the icon to open the context menu.

3. Select "Select > Sequence 1".

   The status indicator in the "Sequence" block is green.
   The status indicators in the "Volatile" block is still red.

3.6.2 Generating an ARB Waveform File

The initial situation is the configuration described in "To assign the sequence to the scenario" on page 27.

To store the created ARB file on the file system of your PC

1. In the repository tree, select "Scenario > Scenario 1".

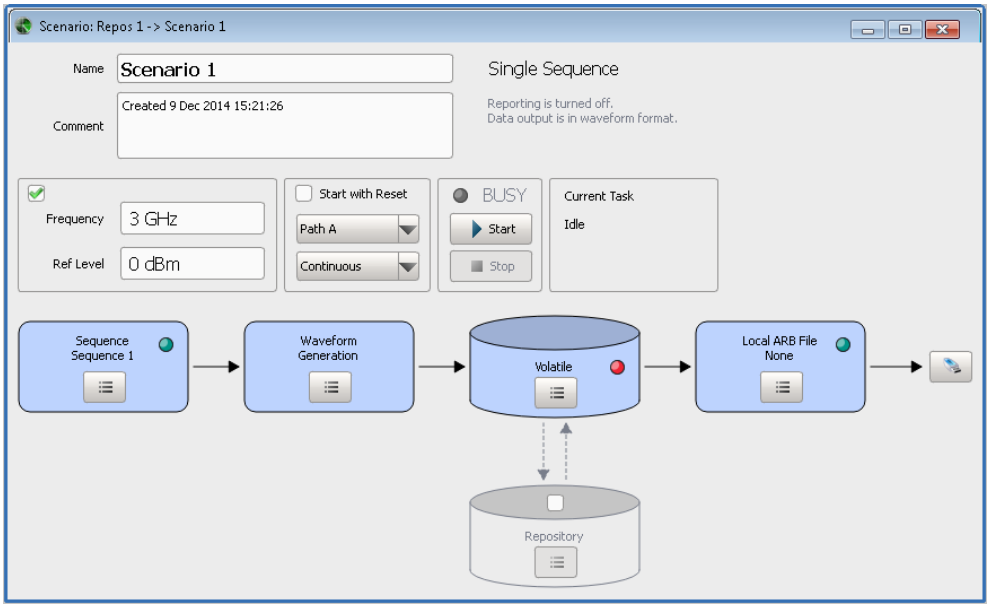
   The "Scenario" dialog opens and displays a block diagram of the signal flow.

2. In the block diagram, select "Local ARB File" and select the icon to open the context menu.

3. In the context menu, select "Set Path...".

4. In the Windows Explorer, navigate to the directory in that the file should be stored.

   The block diagram shows the settings.

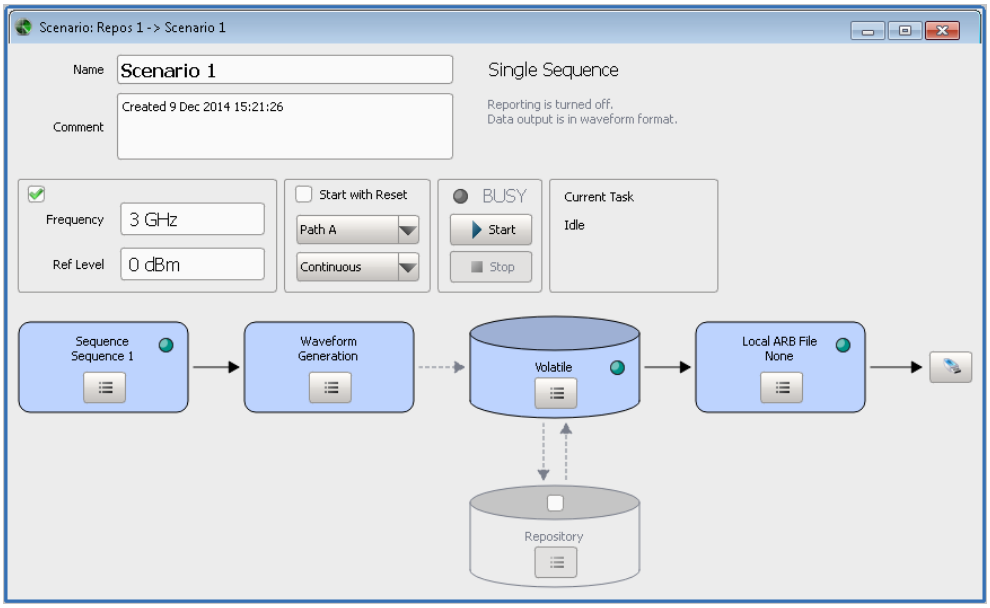


To start the ARB waveform generation

If all preconditions are fulfilled, the "Start" button is active and you can start signal generation.

► In the "Scenario" dialog, select "Start".

The R&S Pulse Sequencer creates the waveform and shows the current progress.



A green indication in the "Volatile" block confirms that the generation is completed.

For information on the provided settings, see:

- chapter 6.2, "Scenario Settings", on page 54

3.6.3 Verifying the Generated Signal in the Waveform View

It is often useful to check the generated waveform visually, before you transfer the waveform to the vector signal generator. The R&S Pulse Sequencer provides a built-in function, the "Waveform View", to represent the generated signal in a graphical form.

The initial situation is the configuration described in "To start the ARB waveform generation" on page 28.

To open the Data View and view the generated signal

► In the "Scenario" dialog, select "Volatile", open the context menu and select "View".

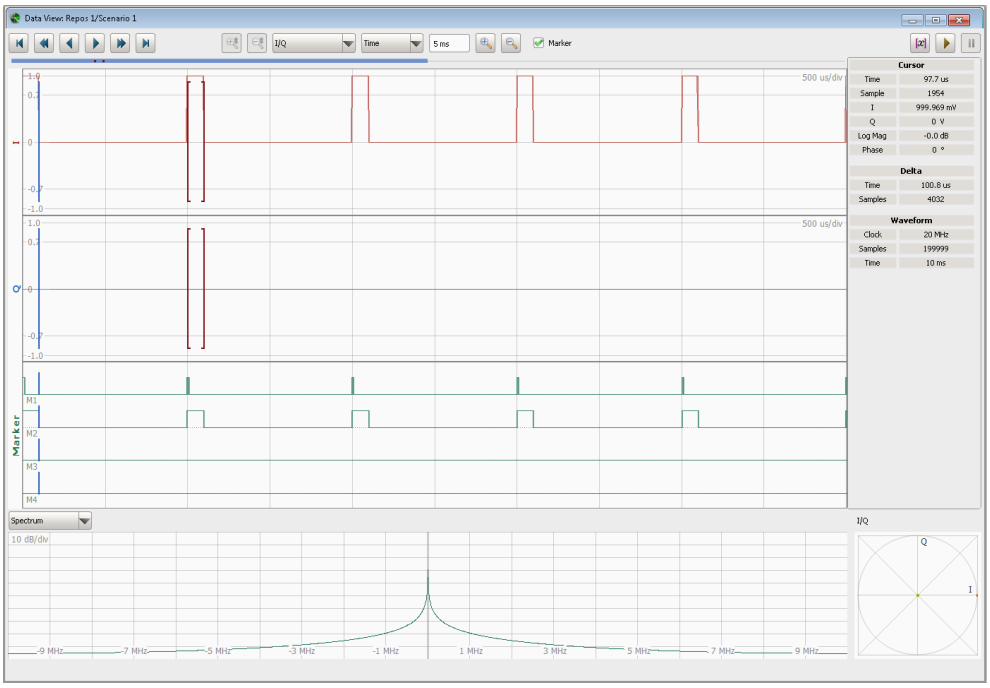


Fig. 3-3: Data View: Sequence composed of 10 unmodulated pulses (first 5 pulses shown)

In its default view mode, the "Data View" dialog displays the I and Q data as a function of time, the spectrum of the generated signal and I/Q the constellation diagram.

The displayed setting confirm that:

- generated is a sequence of 10 pulses but the view port shows only the first 5 pulses
- each pulse has a PRI = 1 ms
- the waveform is 10 ms long, that is 10*PRI
- each pulse has a pulse width of 100 us.

For information on the provided settings, see chapter 18.1, "Waveform and Data View Settings", on page 240.

3.6.4 Launching the Built-In Wizard

To help you get familiar with the software, the R&S Pulse Sequencer provides a built-in startup assistant, the "Wizard".

You may access this "Wizard":

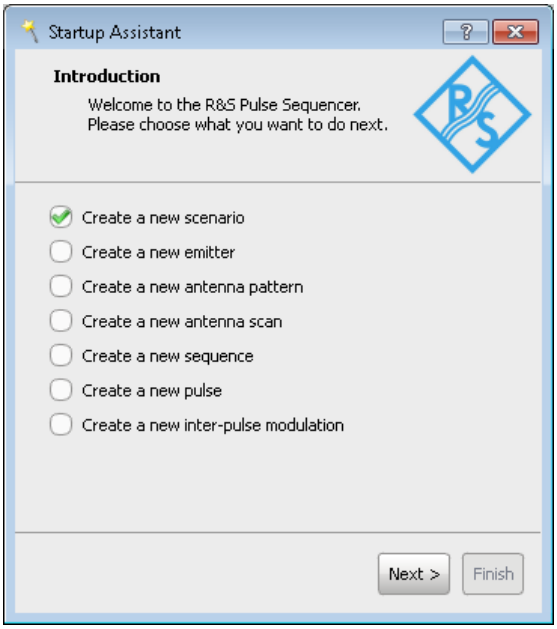
- **On an application start**:
  this is each time you start the software
- **On demand**:

this is any time during operation when you need a fast configuration of just few settings.

To start the built-in wizard

► In the tool bar, select "Help > Wizard".

The "Startup Assistant" opens.



3.6.5 Using the Wizard to Create a Complex Scenario

The main application field of the R&S Pulse Sequencer is the generation of pulsed signals. This example uses the "Wizard" to introduce the way you can create and define antenna patterns, antenna scans, emitters, etc. and interact with the software.

To create a new emitter-based scenario

1. Start the "Wizard", see chapter 3.6.4, "Launching the Built-In Wizard", on page 29.
2. In the assistant dialog, select "Create New Scenario" and confirm with "Next".
3. Select "Complex Scenarios with emitters, modes, and antennas".
4. Follow the instructions and select:
   a) Emitter with: "EIRP = 120 dBW", "Frequency = 3 GHz", "Antenna Pattern > On", "Antenna Scan > ON"
   b) "Antenna Pattern > Gaussian"
   c) "Scan Pattern > Raster"
   d) Pulse with "Rising Edge = Falling Edge = 10 us", "Width = 100 us" and enable "Modulation > On".
   e) "Linear Chirp > On" with "Deviation = 2 MHz"
   f) "PRI = 1 ms" and "Repetition = 5"

5. Confirm with "Finish".

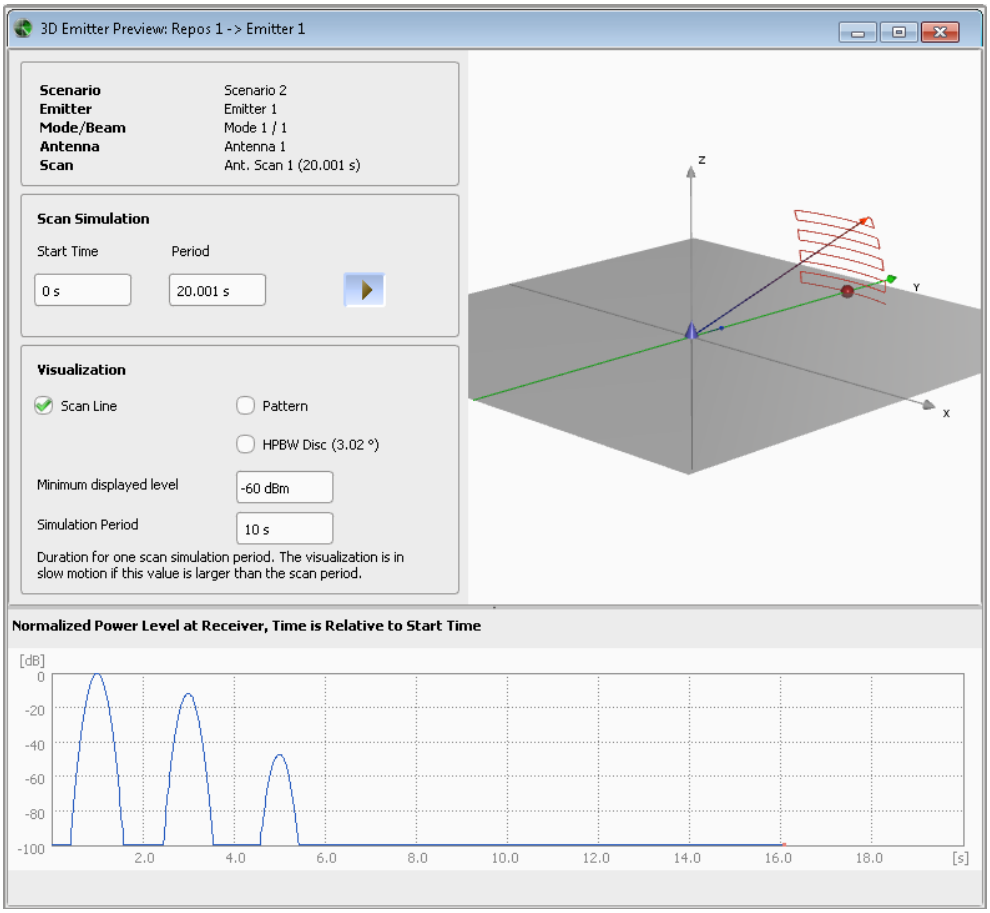
   The software creates the new scenario automatically with all required elements, e.g. one new emitter (`Emitter 1`), one new sequence (`Sequence 2`), new pulse (`Pulse 2`), etc.
   You recognize the created elements by their comment.

   For information on the provided settings, see:

   - chapter 11.1, "Emitter Settings", on page 180
   - chapter 10.2, "Antenna Pattern Settings", on page 155
   - chapter 10.3, "Antenna Scans Settings", on page 164
   - chapter 7.2.5, "Modulation on Pulse (MOP) Settings", on page 79

To visualize the signal received by a static receiver

1. Open the automatically created emitter-based scenario ("Scenario 2").
2. In the "Current Emitter" section on the block diagram, select the "3D" icon.



   The "3D Single Emitter" dialog displays a live plot of the normalized signal power level at the receiver and an interactive 3D view of the emitter antenna pattern and scan.

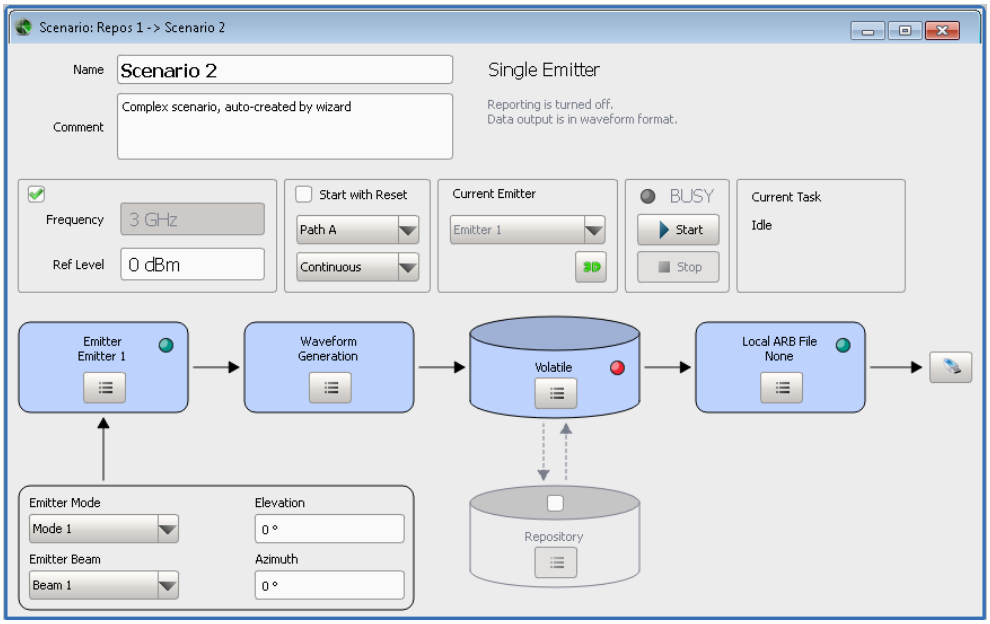
3. Use the mouse wheel to zoom in and out.

4. To rotate the 3D view around its origin, left mouse click, keep the key pressed and move the mouse.

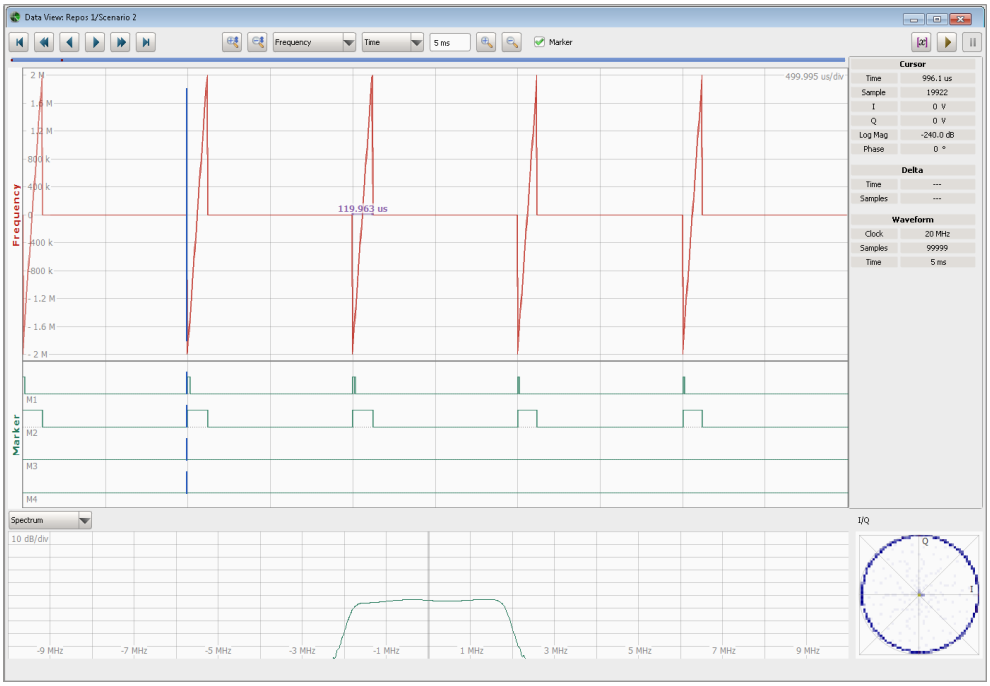
   For information, see chapter 13, "Creating Complex 2D Scenarios with Receivers and Interferers", on page 201

To create and verify the generated ARB waveform

1. In the "Scenario" dialog, define the storage location for the ARB files, see chapter 3.6.2, "Generating an ARB Waveform File", on page 27.



2. Select "Start" to start signal generation.
3. Select "Volatile > View".
4. In the "Waveform View" dialog, select "View Mode > Frequency".
5. Use the "Zoom In" button to change the scale on the y axis.



The "Data View " dialog confirms that the generated sequence consists of 5 chirped pulses, each with a PRI = 1 ms. The frequency variation of each pulse has the typical linear chirp shape.

6. Compare the signal spectrum with the spectrum on figure 3-3.

   You can retrieve more information about the waveform, for example observe the "Waterfall" diagram, display a part of the signal in greater details, enable and display marker traces, etc.

   You find details in the following:

   - chapter 18.1, "Waveform and Data View Settings", on page 240
   - chapter 17, "Defining and Enabling Marker Signals", on page 236

3.6.6 Transferring the Waveform to and Playing it with a Connected Instrument

The example configurations are theoretical cases, because you will not use the software alone. Usually, the PC with the installed software would be connected via LAN to a vector signal generator.

Transferring of waveforms and playing them by a signal generator is beyond the scope of these first steps.

For description, see chapter 19, "Playing the Generated Waveform Files", on page 247.

3.6.7 Saving and Recalling Settings

Repositories are automatically saved when you exist the application. In order to clearly identify new repositories it is advisable to set a meaningful repository name before closing the application.

To rename the repository

Your workspace shows one automatically created repository, "Repos 1".

1. In the repository tree, select "Repository > Repos 1".
2. Change the name and optionally enter a comment.
   E.g. enter "General > Name > K300 and K301 Tests" or a "Comment > Repository with test files".
   To enter a new line in the comment field, press SHIFT+ENTER.
3. Close the dialog.

To save the repository

A yellow data base icon in the repository tree indicates that there are unsaved changes.

► In the menu bar, select "File > Save Repository > K300 and K301 Tests".

The repository is stored in the data path set during the program installation.

To exit the software

► Close the application software.

If the current repository contains unsaved changes, you are prompt to store or ignore them.
Note: Unsaved changes are lost.

For more information, see:

- chapter 3.7.1, "Customizing Your Workspace", on page 35
- chapter 5.3, "How to Manage the Project Data", on page 48

3.6.8 Advanced Features and Examples

With these first steps you have gained an impression of the provided functionality.

You find a comprehensive description of the full range of capabilities in the following sections:

- chapter 7.3, "How to Create a New Pulse and Adjust Its Settings", on page 98
- chapter 8.3, "How to Create Sequences and Use the Control Elements", on page 117
- chapter 6.3, "How to Select and Create a Test Scenario", on page 63
- chapter 14.3, "How to Create Generator Profiles and Configure the Connected Instruments", on page 221

- chapter 18.3, "How to Analyze the Content of Waveform Files and Files with I/Q Data", on page 245
- chapter 19.4, "Playing Waveforms with the Signal Generator", on page 253
- chapter 20.2, "How to Create Test Reports", on page 263
- chapter 10.4, "How to Create a Library with Antenna Patterns and Scans", on page 171
- chapter 11.4, "How to Create and Configure Emitters", on page 186
- chapter 13.2, "How to Create Scenarios with Emitters, Interferers and a Receiver", on page 209
- chapter 12.2, "How to Create a Waveform Scenario and Work with Waveforms", on page 195
- chapter 9.2, "How to Create IPM Profiles and Use Them to Vary Pulse Parameters", on page 144
- chapter 17, "Defining and Enabling Marker Signals", on page 236
- chapter 15.2, "How to Configure the Bit Stream Used by the MOP", on page 230
- "How to import and assign user defined plugins" on page 232

3.7 Customizing the Software

You can customize the R&S Pulse Sequencer's user interface so that it suits to your needs.

- Customizing Your Workspace.....35
- Changing Colors and Default Configuration.....36

3.7.1 Customizing Your Workspace

A workspace comprises the repository tree on the left side with one or more loaded repositories, and the working area with one ore more screens. You can create several workspaces with different repositories assigned to them. You can save and load a user-defined workspace and switch between the different workspaces.

To save a workspace

We assume that you have opened or created a repository and opened one or more dialogs in one of the screens.

1. In the menu bar, select "File > Save Workspace/Save Workspace As" or use the save icon in the toolbar.
2. In the explorer, enter a file name, e.g. `MyWorkspace.pswk` and confirm with "Ok".

   Per default, the `*.pswk` file is stored in the application data directory.

To load a workspace

1. In the menu bar, select "File > Open Workspace".

Existing default and user-defined workspaces are listed.

2. Select a workspace.

   The workspace opens. All dialogs and repositories associated with the workspace are displayed.

To close a workspace

You can close a workspace if you do not need it no longer or you need to switch to a different one.

► In the menu bar, select "File > Clear Workspace".

The workspace is closed.

Closing the workspace unloads all repositories and closes all dialogs.
You can open a workspace again, see "To load a workspace" on page 35.

3.7.2 Changing Colors and Default Configuration

You can change the default application settings, for example, change the display colors or change default settings of the R&S Pulse Sequencer.

To customize the display colors

1. In the menu bar, select "Configure > Colors".



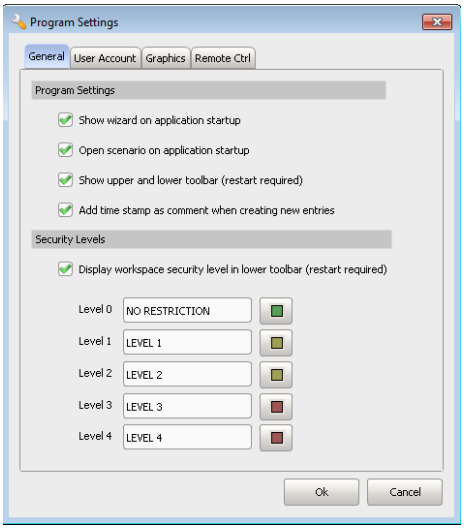
1 = Color scheme
2 = Assigns colors to the different display elements

2. Select a predefined color scheme and if required, adjust the display colors individually.
3. Adjust the display colors of the individual elements, e.g select "Traces > Magnitude" and change the default color.

4. Select "Save As" to store your custom color profile in a file, e.g. `MyColors.col`.

To customize the general program settings

1. In the menu bar, select "Configure > Settings".



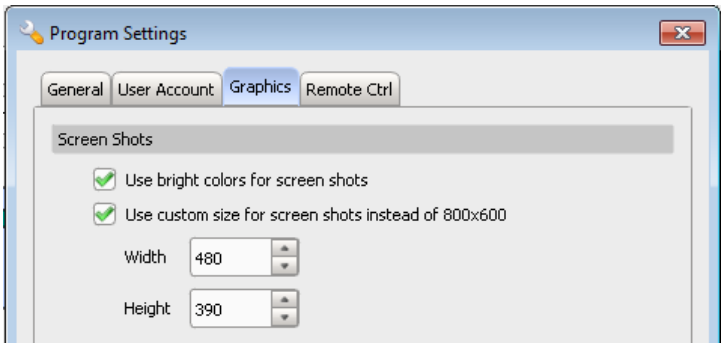
2. Change the settings, for example customize the colors indicating the different security levels.
   For information on the security levels, see "General Repository Settings" on page 42.

3. Confirm with "Ok".

To set the size and color scheme of the hardcopies

For documentation purposes you can create a hardcopy or screen shot of the current screen or dialog. Hardcopies can be sent to the printer or stored to the clipboard.

1. To adjust the hardcopy size, in the menu bar, select "Configure > Settings > Graphics".



2. Select "Use bright colors for screen shots > On" e.g. for ink saving printing.

   Brighter colors are used for all graphical screens instead of the ones set in the current color scheme.

3. Select "Use custom size for screen shots instead of 800x600 > On" and set the "Width" and "Height" in number of pixel.
4. Confirm with "Ok".

The "Program Settings" dialog has two more tabs. The settings provided on these tabs are explained in:

- "To change the default user used to access repositories" on page 52
- chapter 19.3, "Signal Generator Remote Control Settings", on page 252

3.8 Getting Information and Help

If any questions or problems concerning the R&S Pulse Sequencer arise, an extensive online help system is provided in the software and can be consulted at any time.

The integrated help system provides the description of all functions. The help system is context-sensitive and provides information specifically for the current operation or setting to be performed. In addition, general topics provide an overview on complete tasks or function groups as well as background information.

Calling Up Help

► To display the "Help" dialog for the currently focused screen element, e.g. a setting in an opened dialog, select the "Help" icon on the toolbar or press F1.

The "Help" dialog is displayed. A topic containing information about the focused element is displayed. If no context-specific help topic is available, a more general topic or the "Contents" tab is displayed.

The Help toolbar provides buttons with standard navigation functions. The Help window contains several tabs:

- "View" - shows the selected help topic
- "Contents" - contains a table of help contents
- "Index" - contains index entries to search for help topics
  The index is sorted alphabetically. You can browse the list, or search for entries in the list
- "Bookmarks" - contains custom bookmarks
- "Search" - provides text search

To close the Help window

► Select the "Close" icon in the upper right corner of the help window.

4 About the R&S Pulse Sequencer

The R&S Pulse Sequencer generates complex pulsed signal for use with a signal generator. A typical setup includes a PC with installed software and a connected vector signal generator. See figure 4-1 for an example of a simplified test setup.

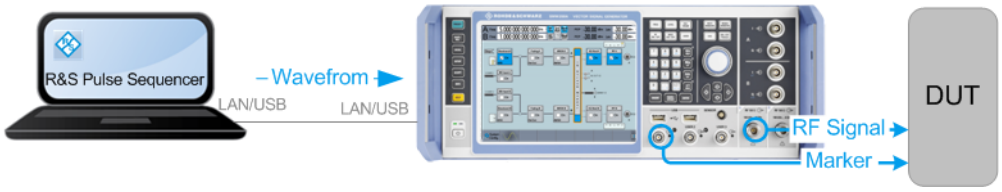


Fig. 4-1: Simplified test setup

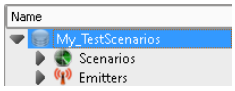
From the configured settings, like selected emitter and receiver configuration, the R&S Pulse Sequencer verifies the capabilities of the connected instrument. The R&S Pulse Sequencer checks, whether the instruments capabilities fulfill the requirements of the generated waveform. In particular, this affects the bandwidth, the carrier frequency and the waveform length of the calculated signal. The R&S Pulse Sequencer then calculates a waveform and transfers it to the instrument.

See also chapter 19, "Playing the Generated Waveform Files", on page 247.

Introduction to the Software Concept

The following is brief introduction to the software concept and the way it handles files and settings.

Repository



The R&S Pulse Sequencer organizes the data in repositories.

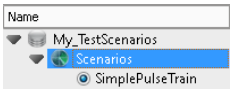
A repository is a file-based database located on the local hard drive or on a network storage drive.

Repositories are entirely managed by the R&S Pulse Sequencer. They can be password protected against unauthorized access. Repositories can be exported as archives and shared among multiple users if they are located on a network drive. Several user can have read permission to load and use the same repository simultaneously. However, only one user can have a write permission at a time.

To move or share repositories with other users, use the "Export/Import Repository Archive" function. Do not directly access the repositories from your windows Explorer.

For more information, see chapter 5, "Organizing the Project Data in Repositories", on page 42.

Scenario



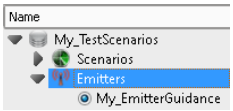
The scenario is the top level description of the signals to be generated.

The type of the scenario determines which kind of signal is calculated and how this signal is processed. A more complex scenario combines several signals. Scenarios may also contain information about the time variation of the signals.

Starting a scenario creates the I/Q waveform, sends them to the ARBs and configures the signal generator(s).

For more information, see chapter 6, "Selecting a Suitable Scenario and Creating Scenarios", on page 53.

Emitter



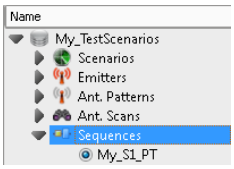
Emitters emulate radar systems. The emitter combines a sequence, an antenna scan, and an antenna pattern to one logical unit.

An emitter can work in more than one operating mode and switch between them. The operating mode is a description of the radar purpose, such as scanning, searching or tracking. For example, an airport surveillance radar can switch between different scan types to observe its airspace.

Each mode can have individual antenna and signal configuration. Multiple beams are possible within a mode.

For more information, see chapter 13, "Creating Complex 2D Scenarios with Receivers and Interferers", on page 201.

Sequence

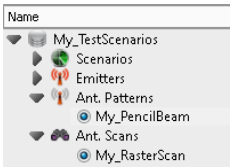


A sequence describes how pulses are arranged to form a waveform. The most simple sequence comprises one pulse that is repeated with a constant PRI (pulse repetition interval). Typical sequences however are rather complex; they may contain pulse definitions, waveforms, and sequencing elements such as repetitions and loops.

The sequence is a logical description. It cannot directly be represented as an I/Q waveform.

For more information, see chapter 8, "Building Pulse Sequences", on page 106.

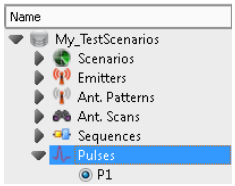
Antenna



The antenna characterizes the radiation pattern of the emitter. Typical radar antennas like the phase array antennas can adapt their radiation pattern dynamically, depending on the operating mode the radar is working in. For example, a antenna can use a cosecant beam in scan mode and pencil beam in track mode.

For more information, see chapter 10, "Defining Antenna Patterns and Antenna Scans", on page 154.

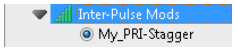
Pulse



In the R&S Pulse Sequencer, pulses are the fundamental building block of any signal. Pulses are described mathematically with their amplitude (envelope), the applied modulation on pulse (MOP) and the enabled marker signals. One single pulse description is however not sufficient to calculate (and process) a signal; pulses have to be organized in a sequence, and assigned to an emitter.

For more information, see chapter 7, "Creating a Pulse Library", on page 66.

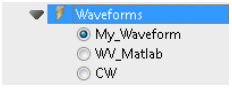
IPM (Inter-Pulse Modulation)



Inter-Pulse Modulation (IPM) profiles modify a pulse parameter (like level, pulse repetition interval, frequency offset, etc.) from one pulse to another one. The IPM profiles are applied whitin a sequence.

For more information, see chapter 9, "Defining and Enabling Inter-Pulse Modulation Effects", on page 127.

Waveforms and Interferers

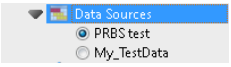


Additionally to creating pulse signals you can also load waveform files and create waveform scenarios with them or even use them as interferes.

An interferer contains complete waveforms, that can be user defined or created with the signal generation software R&S WinIQSIM2. Interferes can be mixed with the signal.

For more information, see chapter 12, "Working with Waveforms and Generating Interfering Signals", on page 192.

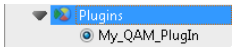
Data Source



Data source defines a numerical source for the symbols used in the modulations applied on the pulse (MOP).

For more information, see chapter 15, "Defining the Data Source", on page 226.

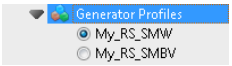
Plugins



A plugin is an extension to the standard functions of the R&S Pulse Sequencer. For example, a plugin can be a user-defined MOP or IPM profile. For even more flexibility, plugins may contain variables. If different values are assigned the same variable in the different pulses, the applied MOP varies.

For more information, see "How to import and assign user defined plugins" on page 232.

Instruments and Generator Profiles



The generator profile is a logical placeholder for a physical instrument. It describes the minimum requirements that must be fulfilled so that the signal can be generated. Generator profiles are assigned to physical instruments.

For more information, see chapter 14, "Creating Generator Profiles and Configuring the Instruments", on page 216.

See also chapter 19, "Playing the Generated Waveform Files", on page 247

5 Organizing the Project Data in Repositories

This section explains the basic repository concept and informs you briefly how the software data base is built.

An introduction to the concept of elements handling is provided in "Creating elements" on page 23.

5.1 Repository Settings

The repository tree shows the content of the currently loaded repositories.

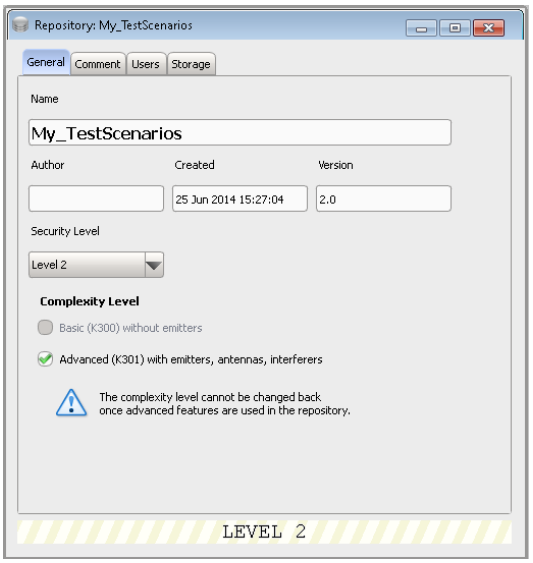
To access the repository settings:

► In the project tree, double click on the repository name.

Available are the following settings:

General Repository Settings....42
Comment....43
Users Management....43
Storage....44

General Repository Settings
The "General Settings" tab provides general information, like the name of the author and the date of creation.



"Name" When a new repository is created, it is named automatically but you can rename it afterwards.
The automatically assigned names follow the syntax:
`New_<yyyy-mm-dd><T><hhmmss>`, where `<yyyy-mm-dd>` and `T<hhmmss>` indicate the current date and time respectively.

"Security Level"
A security level information is assigned to each repository, where "Level 0" is the lowest level that means no restrictions and "Level 4" is the highest one. Elements belonging to a repository with higher security level cannot be copied to a repository with lower one.
If more than one repositories are currently loaded, the workspace displays the highest required security level, see chapter 3.4, "Understanding the Displayed Information", on page 19.

"Complexity Level"
The complexity level defines whether basic or advanced features are used in the repository. The complexity level also reflects the option concept of the signal generators.
For example, if "Complexity Level > Advanced (K301)" is used, the signal generator has to be equipped with the option Advanced Features for Pulse Sequencer, e.g. R&S SMW-K301.

See also:

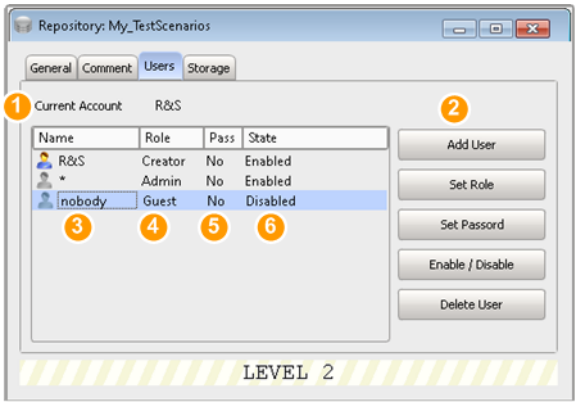
- chapter 2, "Welcome to the R&S Pulse Sequencer", on page 13
- chapter 3.1, "Required Options", on page 15

Remote command:
`REPository:CATalog?` on page 276
`REPository:CREate` on page 277
`REPository:SELect` on page 277
`REPository:REMove` on page 279
`REPository:AUTHor` on page 352
`REPository:DATE` on page 352
`REPository:VERSion` on page 353
`REPository:COMPlexity` on page 352
`REPository:SECurity` on page 353

Comment
Enter a short description of the repository.

Remote command:
`REPository:COMMent` on page 278

Users Management
The R&S Pulse Sequencer uses its *own user rights management* that is independent from the user administration of the operating system but follows the same principles: the current user cannot delete its own account; users with lower access rights cannot delete accounts with higher rights.



1 = Current user
2 = Standard functions for user management
3 = Login name or custom user name
4 = Assigned user role; the creator role is assigned automatically and cannot be changed (see also table 5-1)
5 = Identifies if the repository is password protected or not
6 = Current user state

Repositories support a simultaneous access from several users. Per default, all users are granted with the same (administrator) rights. The user rights can be restricted by assigning of a different user role, other than the creator role. At least one user with administrator rights must always exist.

To protect a repository form accidental changes, the repository data can be password protected. Passwords are then required to access a repository and any loss of the password would render the repository inaccessible.

Table 5-1: Overview of the available user roles and their access rights

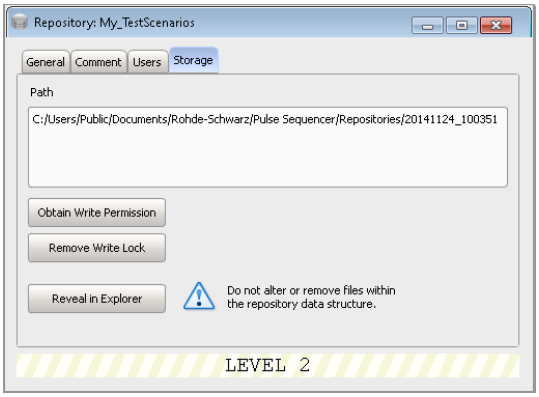
User role	Read access	Write access	Create a scenario form the existing elements	Delete elements	Copy repository elements
Creator	x	x	x	x	x
Admin	x	x	x	x	x
User	x	-	x	-	x
Guest	x	-	-	-	x

Per default, the R&S Pulse Sequencer uses your Windows login name to access the repositories but you can also use a custom name.

See also:
- "To protect a repository from accidental changes" on page 51
- "To change the default user used to access repositories" on page 52

Remote command:
`REPository:ACCess?` on page 351

Storage
This tab provides settings necessary to configure the default storage folder for the repositories.



"Path" Displays the location (network or local folder) the repository is currently stored in.

Remote command:
`REPository:PATH?` on page 353

"Obtain Write Permission"

Repositories allow the simultaneous access from several users. If a user with write permission has opened a repository, this write permission is exclusive. Another user will be granted with the write permission only after the first user had closed the repository or removed it from its workspace.
You can request an exclusive write permission only on a repository that is currently not opened by another user with write permission.

See also:

- "Discovered Repositories on the Mass Storage" on page 47
- "To obtain write permission on a repository" on page 50

"Remove Write Lock"

The R&S Pulse Sequencer locks an improperly or unexpectedly closed repository and indicates this situation with a red lock symbol in the *"Repository Tree"*. A locked repository can be unlocked only from the same PC and by the same user with write permission that had opened the repository before.
See "To remove write lock that results from a previously crashed session" on page 389

"Reveal in Explorer"

You can open the repository file structure in the Windows Explorer.

Note: Do not change the repository content by changing the file structure in the windows Explorer!
To move or share repositories with other users, always use the "Export/Import Repository Archive" function.

See:

- "To create and export a repository archive" on page 50
- "To import a repository archive" on page 50

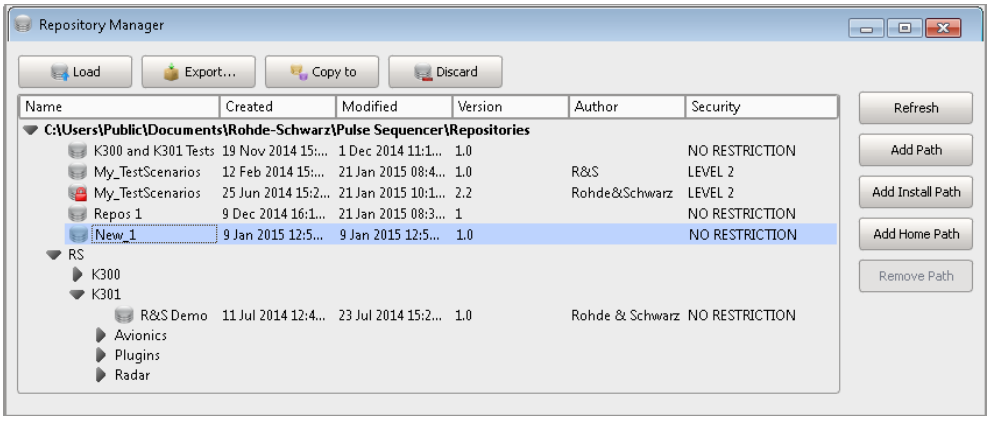
Remote command:
`REPository:FILename?` on page 352
`REPository:SAVE` on page 353

5.2 Repository Manager Settings

The "Repository Manager" comprises the setting for managing the repository files and archives on the mass storage.

To access the "Repository Manager":

► In the menu bar, select "File > Load and Manage Repository".



Available are the following settings:

Discovered Repositories on the Mass Storage....47
Load....47
Export....47
Copy to....47
Discard....47
Refresh....47
Add Path, Add Install Path, Add Home Path....47
Remove Path....48

Discovered Repositories on the Mass Storage
Lists all found directories and the repository files they contain.

You can change the column width and order.

A red lock symbol on the data base icon in the "Repository Manager" indicates one of the following situations:
- repository is currently opened by another user with explicit write access
- insufficient user right for the particular repository
- insufficient user right on the file system (directory) the repository is stored in.

Try out to:
- wait until the repository has been closed and try to obtain exclusive write permission, see "To obtain write permission on a repository" on page 50.
- change your current user or use a user with different role, see "To change the default user used to access repositories" on page 52
- login as user with sufficient (write) access rights on the file system.

Remote command:
`REPManager:CATalog?` on page 353

Load
Loads and displays the selected repository in the workspace.

See "To load a repository" on page 49.

Remote command:
`REPManager:LOAD` on page 354

Export
Exports the selected repository to an archive file.

See "To create and export a repository archive" on page 50.

Remote command:
`REPManager:EXPort` on page 355

Copy to
Copies the selected repository to a new location.

The R&S Pulse Sequencer will automatically copies all relevant files and data base elements.

Discard
Discards the entire repository from the permanent mass storage, if:
- the current user has write permission, see table 5-1
- the repository is not opened by another user with write permission

Remote command:
`REPManager:DISCard` on page 354

Refresh
Refreshes the list of found repository files.

Add Path, Add Install Path, Add Home Path
Opens the "Find Path" dialog that is very similar to the Windows Explorer.

Navigate to the required directory and load it. Found repository files are displayed in the list.

Remote command:
`REPManager:PATH:ADD` on page 355
`REPManager:CATalog?` on page 353

Remove Path
Removes the selected path. The repository files within this directory are also removed from the list.

Remote command:
`REPManager:PATH:DELete` on page 356

5.3 How to Manage the Project Data

See:

- "To create a new repository" on page 48
- "To rename the repository" on page 34
- "To save the repository" on page 34
- "To delete a repository" on page 49
- "To remove a repository from the workspace" on page 49
- "To load a repository" on page 49
- "To create and export a repository archive" on page 50
- "To import a repository archive" on page 50
- "To share a repository with other users" on page 50
- "To obtain write permission on a repository" on page 50
- "To protect a repository from accidental changes" on page 51
- "To remove the password protection on a repository" on page 52

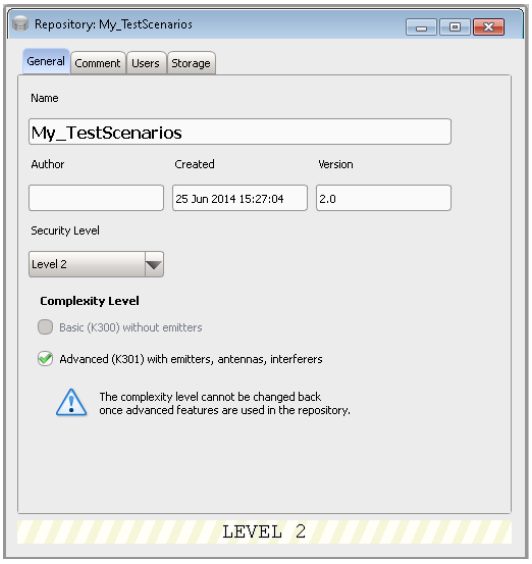
To create a new repository

1. In the menu bar, select "File > New Repository".

   A new empty repository is created and added to the project tree.
   Repositories are named automatically. The repository names follow the syntax: `New_<yyyy-mm-dd><T><hhmmss>`, where `<yyyy-mm-dd>` and `T<hhmmss>` indicate the current date and time respectively.

2. In the project tree, double click on the repository name.

   The "Repository" dialog opens.



3. Change the settings, e.g. change the automatically assigned "Name" or add a "Comment".
4. Set the "Complexity Level > Advanced (K301)".
5. Close the dialog.

To delete a repository

You can delete repositories that you no longer need, if this repository is not opened by another user with write permission.

1. Use a user with write access rights, e.g. Admin or Creator (see table 5-1).
2. In the menu bar, select "File > Load and Manage Repository".
3. From the listed files, select a repository file, e.g. "My_TestScenarios".
4. Select "Discard".

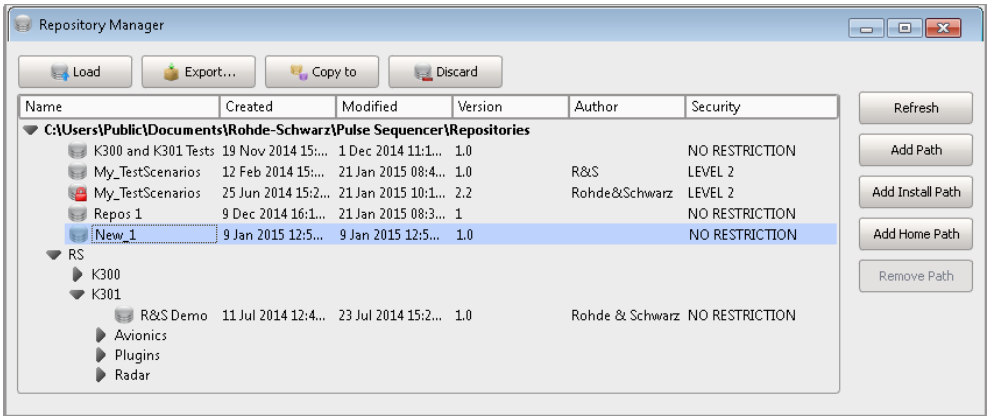
To remove a repository from the workspace

► In the project tree, select the repository e.g. "My_TestScenarios", open the context menu and select "Remove from Workspace".

The repository is removed form the workspace but not deleted. You can open it again, see "To load a repository" on page 49.

To load a repository

1. In the menu bar, select "File > Load and Manage Repository".
2. If necessary, for example by loading and exchanging repositories with another user, change the directory the repository files are searched in.
3. From the listed files, select a repository file, e.g. "New_1".



4. Confirm with "Load".

   The repository is opened and displayed in the project tree.

To create and export a repository archive

We assume that you have created a repository and configured at least one scenario.

1. In the menu bar, select "File > Load and Manage Repository".
2. If necessary, change the directory repository archive files are stored in.
3. From the listed files, select a repository file, e.g. "My_TestScenarios".
4. Select "Export".

   The repository archive is stored in a file with extension `*.psarch` .

To import a repository archive

1. In the menu bar, select "File > Import Repository".
2. Navigate to the network directory and select a `*.psarch` file.

   The "Load and Manage Repository" dialog opens.
3. Select the a repository file and confirm with "Load".

   The repository is opened and displayed in the project tree.

To share a repository with other users

To move or share repositories, use the "Export/Import Repository Archive" function. Do not access the repositories from your Windows Explorer.

1. See "To create and export a repository archive" on page 50
2. See "To import a repository archive" on page 50

To obtain write permission on a repository

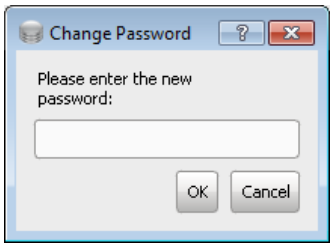
Repositories can be opened with executive write permission by one user at a time.

If you have sufficient user rights, you can request an exclusive write permission on a repository that is currently not opened by another user with write permission.

1. In the project tree, double click on the repository name.
2. In the "Repository" dialog, select "Storage".
3. Select "Obtain Write Permission".

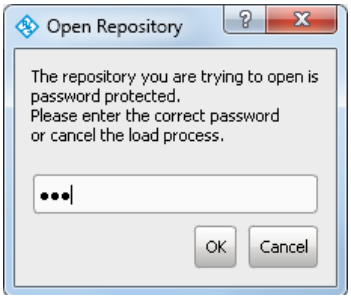
To protect a repository from accidental changes

1. In the project tree, double click on the repository name.
2. In the "Repository" dialog, select "Users".
3. To protect the repository with a password:
   a) Select "Users > <current Account>" and select "Set Password".
   The "Change Password" dialog opens.

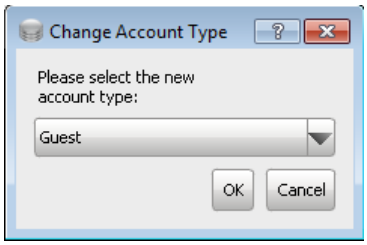


   b) Enter a password and confirm with "Ok".

   On a new start up or when this repository is loaded, the R&S Pulse Sequencer requires the correct password to load the repository.


Open Repository
The repository you are trying to open is password protected.
Please enter the correct password or cancel the load process.
OK
Cancel

4. To allow a read-only access:
   a) Select "Users > *" and select "Set Role".
   The "Change Account Type" dialog opens.



   b) Select a new role with read-only access, e.g. "Guest" and confirm with "Ok".

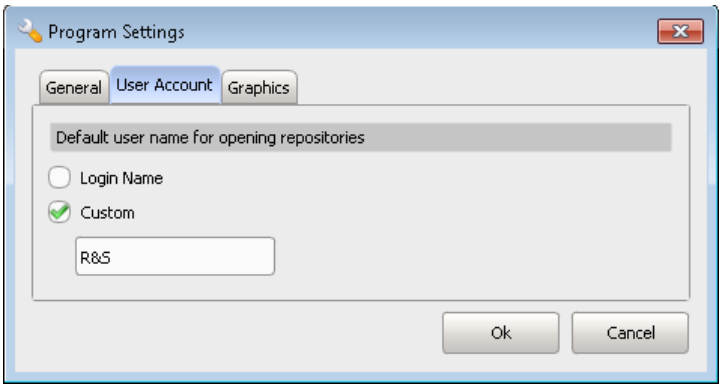
To remove the password protection on a repository

1. In the project tree, double click on the repository name.
2. In the "Repository" dialog, select "Users".
3. Select "Users > <current Account>" and select "Set Password".
4. In the "Change Password" dialog do not enter any password.
5. Confirm with "Ok".

   The "Users" dialog confirm that the repository is not protected with password.

To change the default user used to access repositories

1. In the toolbar, select "Configure > Settings > User Account".



2. Select "Custom > On" and enter an existing user name.

6 Selecting a Suitable Scenario and Creating Scenarios

This section explains the provided scenario types and helps you select the scenario most fitting to your current task.

6.1 Overview of the Available Scenarios and Their Complexity

The R&S Pulse Sequencer supports six scenarios with different complexity. Select the scenario most fitting your test case:

- **Single Sequence**
  The Single Sequence scenario is used to create sequences from individual pulses. Pulse parameters include the pulse envelope, modulation on pulse (MOP), carrier frequency, and level. The sequence editor defines the order in which the pulses are generated and also adds control elements such as loops and fillers. Inter pulse modulation profiles can be applied to vary pulse parameters.
- **Sequences (Collection)**
- The Sequences (Collection) scenario is used to create multiple sequences which can be selected in arbitrary order.
  Pulse parameters include the pulse envelope, modulation on pulse (MOP), carrier frequency, and level. The sequence editor defines the order in which the pulses are generated and also adds control elements such as loops and fillers. Inter pulse modulation profiles can be applied to vary pulse parameters.
- **Waveform Sequence**
  The Waveform Sequence combines individual waveforms to one joint waveform or to one multi-segment waveform. The sequence editor defines the order of the individual waveforms and adds control structures such as loops and fillers.
- **Single Emitter**
  The Emitter scenario combines a pulse train with an antenna pattern and a particular antenna scan type. An emitter comprises several modes. Each mode is defined by an antenna pattern and an antenna scan as well as multiple beams that contain the pulse train.
- **Emitters (Collection)**
  The Emitter Collection scenario provides a choice of multiple emitters that can be selected manually. An emitter comprises several modes. Each mode is defined by an antenna pattern and an antenna scan as well as multiple beams that contain the pulse train.
- **Localized Emitters**
  The Localized Emitters scenario combines multiple emitters and one receiver on a 2D map. An emitter comprises several modes. Each mode is defined by an antenna pattern and an antenna scan as well as multiple beams that contain the pulse train. The receiver is defined by a single antenna pattern and scan.

6.2 Scenario Settings

To access these settings:

► Select "Repository Tree > Scenario".

The "Scenario" dialog provides the settings, necessary to configure the signal processing, that is to select the signal source, to create the ARB waveform, to select the generator profile and to transfer the waveform. The dialog provides also basic instrument control functions that are required for waveform playback and signal routing.
The dialog also shows an interactive block diagram of the main processing blocks, together with their current settings and state.

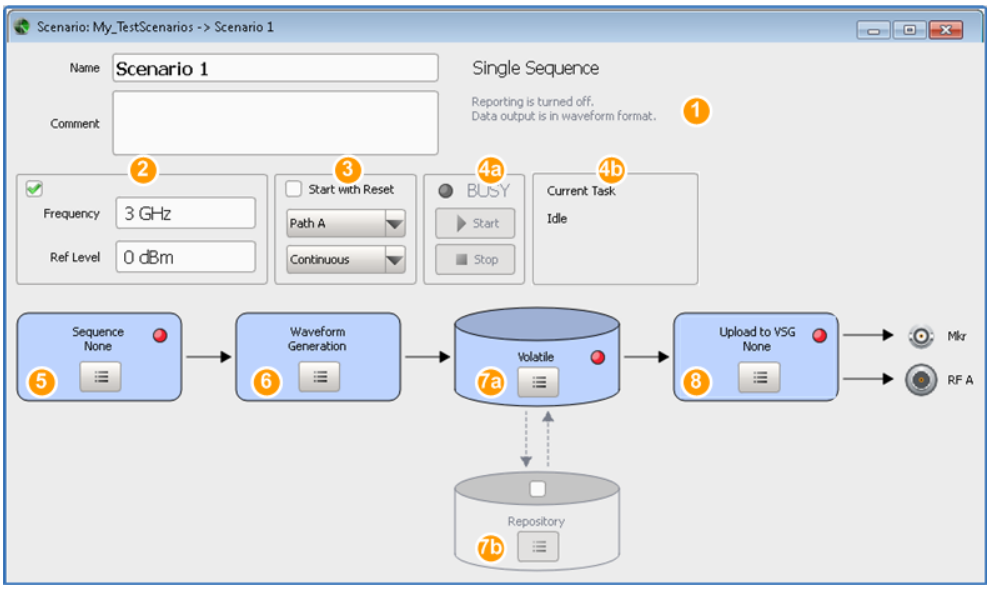


Fig. 6-1: Scenario dialog: Understanding the displayed information

1 = Status information: report generation and kind of the generated waveform file
2 = Major RF parameters
3 = Functions for basic instrument control
4a = Start builds the ARB waveform and transfers it to the instrument
4b = Waveform generation status, displayed during waveform calculation
5+6+7+8 = Block diagram of the processing chain
5 = Represents the signal source the ARB waveform file is built up from, i.e. single or multiple pulse sequences, a waveform sequence or single or multiple emitters
6 = Accesses settings that influence the Waveform Generation
7a, 7b = Volatile, Repository represent the temporary memory and the network storage the generated ARB waveform is stored in
8 = Displays the currently selected generator's profile (Upload to VSG (Vector Signal Generator)) or indicates that a local ARB waveform is generated (Load ARB File)
Green LED = Indicates suitable source and instrument profile or stored local waveform
Yellow LED = Indicates insufficient instruments capabilities
Red LED = Indicates that there is no source or instrument's profile or no waveform assigned

The provided settings depend on the selected scenario type.

6.2.1 Common Scenario Settings

The following settings are common to all scenarios:

Scenario Name 55
Comment 55
Scenario Type 55
Status information 55
Set (RF level and frequency on target instrument) 56
Frequency 56
Ref. Level 56
Start with Reset 56
Path 57
Playback mode 57
Start/Stop, Busy 57
Waveform Generation 57
Volatile, Repository 57
Upload to VSG (Vector Signal Generator) 58
Load ARB File 59
New, Edit, Clear, Select, Config, View 59

Scenario Name
Enter an unique name to identify for your current scenario in the repository.

Remote command:
`SCENario:CREate` on page 277
`SCENario:CATalog?` on page 276
`SCENario:NAME` on page 278
`SCENario:SELect` on page 277
`SCENario:REMove` on page 279

Comment
Enter a short description of the scenario. The first line is displayed in the repository tree.

Remote command:
`SCENario:COMMent` on page 278

Scenario Type
Displays the scenario type.

Remote command:
`SCENario:TYPE` on page 361

Status information
Displays information on:

- kind of generated waveform file, single segment or MSW
- report generation status, type and file location.
  To enable report generation, select "Waveform Generation > Reporting > Reporting > Enable"
  See also chapter 20, "Creating Reports and Documenting Measurement Results", on page 260.

Set (RF level and frequency on target instrument)
If enabled, the RF frequency and output level on the signal generator will be automatically configured.

Remote command:
`SCENario:OUTPut:RF:ENABle` on page 371

Frequency
Sets the carrier frequency of the instrument on which the generated waveform is modulated on.

Emitters use their current operating frequency, see "EIRP" on page 181.

Remote command:
`SCENario:OUTPut:FREQuency` on page 370

Ref. Level
Sets the reference level.

This value is used by the calculation of the pulse envelope. The reference level corresponds to the "Ref." line, displayed on the "Pulse Envelope Graph" as a relative ratio. For example, "Ref. = 1" corresponds to 100% * "Ref. Level".

For unattenuated pulses, the "Ref. Level" sets the pulse top power.

Example: Pulse top power of an unattenuated pulse

If:
- "Ref. Level = -30 dBm"
- "Pulse > Level > Attenuation > Top Power = 0 dB"

Then the pulse top power would be -30 dB (at the RF output).

Example: Pulse top power of an attenuated pulse

If:
- "Ref. Level = -30 dBm"
- "Pulse > Level > Attenuation Top Power = 10 dB"

Then the pulse top power would be -40 dB (at the RF output).

If:
- "Ref. Level = -30 dBm"
- "Pulse > Level > Attenuation > Top Power = 0 dB"
- Δ Level = 10 dB

Then the pulse top power would be -20 dB (at the RF output).

Remote command:
`SCENario:OUTPut:LEVel` on page 370

Start with Reset
Sends an initial reset command (`*RST`) to the connected instrument to set this instrument to a definite state.

Remote command:
`SCENario:OUTPut:RESet:ENABle` on page 371

Path
If the connected instrument is a two-path instrument, selects the signal path that will process the generated signal.

Remote command:
`SCENario:GENerator:PATH` on page 362

Playback mode
Defines the way the generated sequence is processed: played back once ("Single Shot") or looped ("Continuous").

Remote command:
`SCENario:OUTPut:RUNMode` on page 371

Start/Stop, Busy
Standard control function.

If the prerequisites are fulfilled, the "Start" button is active.

Select "Start" to start signal calculation. The software informs you about the estimated file size. A red "Busy" indication and a detailed progress information indicate that the calculation is in progress.

The waveform is transferred to and processed by the connected instrument. Signal routing and RF settings are configured as defined with:
- Set (RF level and frequency on target instrument), Frequency and Ref. Level
- Start with Reset, "Path" on page 57 and "Playback mode" on page 57

Remote command:
`SCENario:STARt` on page 361
`SCENario:STOP` on page 361
`SYSTem:PROGress?` on page 383
`SCENario:STATe?` on page 361

Waveform Generation
In the general case, the software calculates a waveform, most suitable for the current signal generator; e.g. an ARB file or a multi segment waveform.

You can change the settings influencing the waveform generation, see chapter 19.1, "Waveform Generation Settings", on page 247.

Volatile, Repository
Represents the volatile (temporary) memory and the repository.

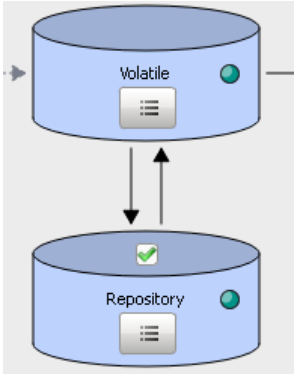
Depending on your user rights, you have read-only or a full access to the repository. Users granted with full access rights are allowed to perform:

"Volatile > View"
Visualizes the generated waveform file from the volatile memory. See chapter 18, "Visualizing and Analyzing Waveforms", on page 240.

Remote command:
`SCENario:VOLatile:VIEW` on page 372

"Repository > On"
Enables file storage in the repository.

"Volatile > Release to Repository"
Exports and stores the waveform files in the repository, i.e. the waveform is archived in the repository.
You can access archived waveforms any time latter, e.g. after software restart.



"Volatile > Load from Repository"
Loads and play files from this storage; stored waveform files are played back without being recalculated.

"Volatile/Repository > Clear"
Deletes files from the volatile memory and the repository.

Remote command:
`SCENario:CACHe:VOLatile:VALid?` on page 362
`SCENario:CACHe:VOLatile:CLEar` on page 362
`SCENario:OUTPut:REPository:ENABle` on page 371

Upload to VSG (Vector Signal Generator)
Select "Upload to VSG > Target > Generator" to transfer and upload the waveform(s) to a signal generator. The second line indicates the currently select generator profile.

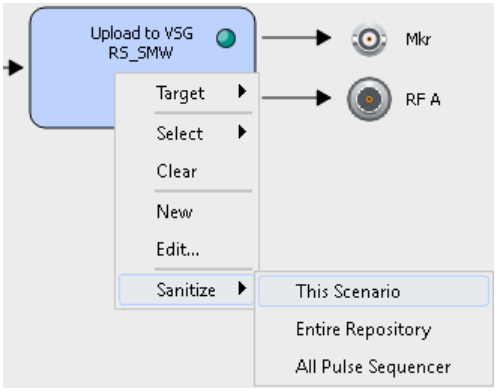


To assign a generator profile, select "Upload to VSG > Select" and select an existing profile.

Profiles must match with the capabilities of a physical instrument:
- **Green LED**: indicates that the selected profile is suitable for the current scenario; If all other prerequisites are fulfilled, the "Start" function is active.
- **Yellow LED**: indicates insufficient capabilities
- **Red LED**: indicates that no profile is assigned

In a "Scenario Type > Localized Emitters", use the "Assign Emitters" function to perform a signal to instrument mapping.

With the "Sanitize" function, you can remove uploaded waveforms from the hard disk of the signal generator.



See also:

- chapter 19.2, "Signal to Generator Mapping Settings", on page 250
- "To retrieve the profile of the connected generator" on page 224
- "To find connected instruments and assign the generator profiles to them" on page 222
- "To assign a connected instrument to a scenario" on page 225
- "To map the generated signals to the signal generator" on page 214
- "To delete wavefroms form the hard disk of the signal generator" on page 255

Remote command:

`SCENario:OUTPut:TARGet` on page 372

`SCENario:GENerator` on page 362

`SCENario:GENerator:PATH` on page 362

Load ARB File

Select "Upload to VSG/Local ARB File > Target > File" to generate an ARB file instead of transferring the file directly to the signal generator.



- Use the "Load to ARB > Select > Set Path" function to define the storage place for the generated ARB file.
- Use the "Load to ARB > Select > Open Path in Explorer" function to access the storage folder in the Windows Explorer.

Remote command:

`SCENario:OUTPut:PATH` on page 371

New, Edit, Clear, Select, Config, View

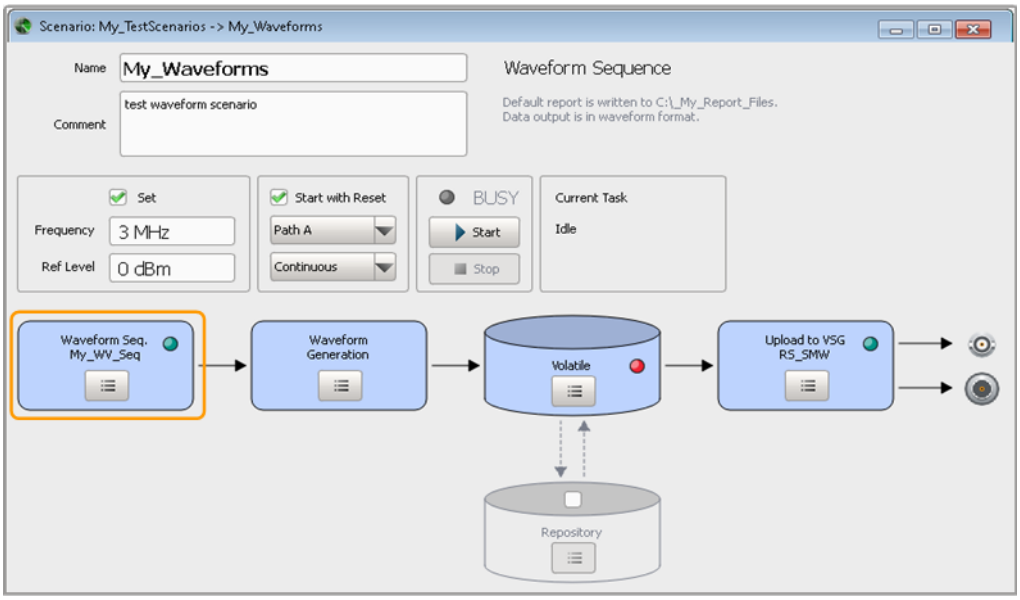
Standard functions in the context-sensitive menu.

See table 3-4.

6.2.2 Scenario Specific Settings

Single Sequence and Waveform Scenario Settings..... 60
Sequence (Collection) Scenario Settings.....61
Single Emitter Scenario Settings.....61
Emitters (Collection) Scenario Settings.....62
Localized Emitters Scenario Settings.....63

Single Sequence and Waveform Scenario Settings



(for an example of a single sequence scenario, see figure 6-1)

These scenarios generate an ARB signal form a selected sequence or a waveform.

Use the context menu in the "Sequence/Waveform" block to select and configure the transmitted sequence or waveform.

See:

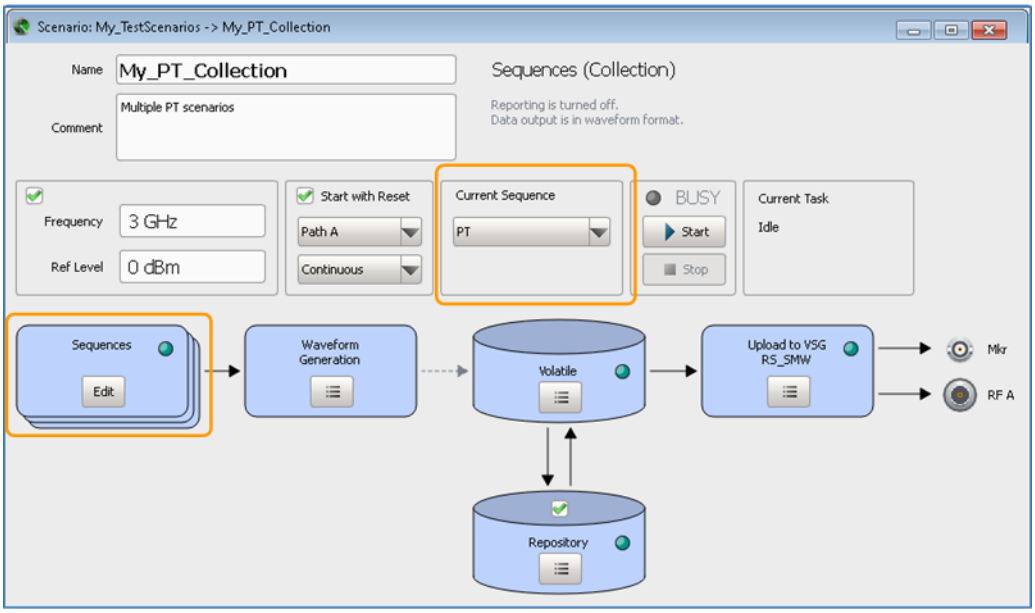
- chapter 8, "Building Pulse Sequences", on page 106
- chapter 12, "Working with Waveforms and Generating Interfering Signals", on page 192
- chapter 19.4.1, "How to Create, Transfer and Play the Waveforms of Simple Scenarios", on page 253
- chapter 6.2.1, "Common Scenario Settings", on page 55

Remote command:

`SCENario:SEQuence` on page 363

`SCENario:SEQuence:CLEar` on page 363

Sequence (Collection) Scenario Settings



A scenario with multiple sequences requires a list of sequences. You can switch between these sequences and select the one to be transmitted.

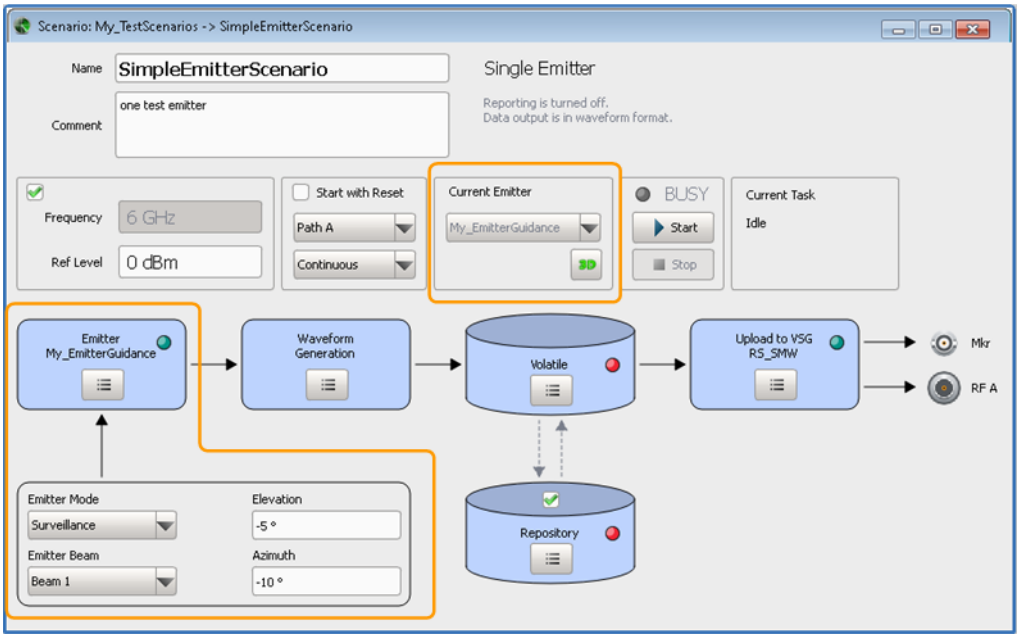
The "Current Sequence" indicates the alias name of the currently selected one.

See:

- chapter 8.2.6, "Lists with Multiple Sequences", on page 116
- chapter 6.2.1, "Common Scenario Settings", on page 55

Remote command:
`SCENario:CSEQuence` on page 368

Single Emitter Scenario Settings



Use the context menu in the "Emitter" block to select and configure the emitter.

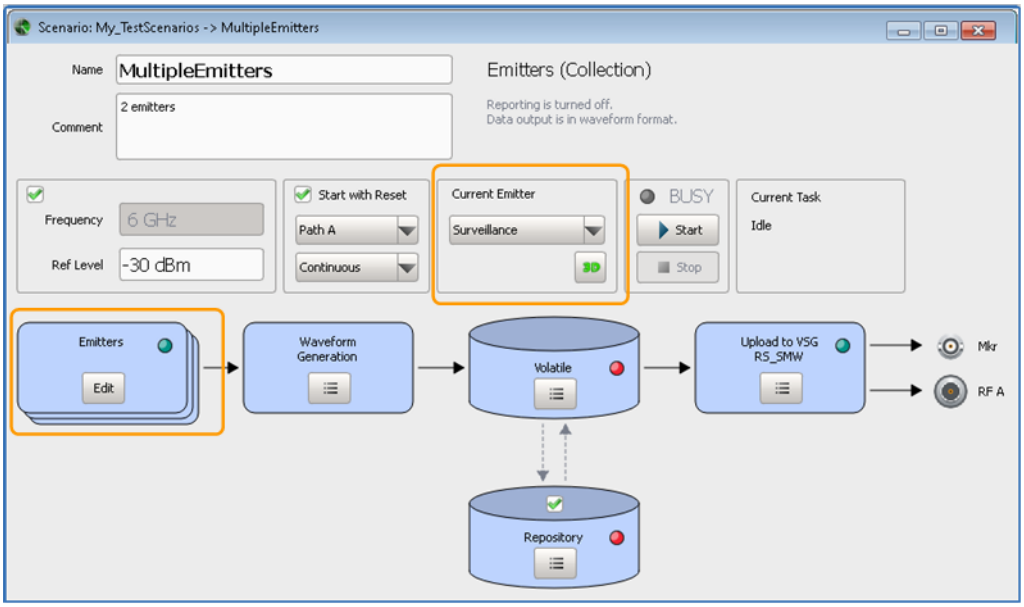
You can switch between the emitter modes, define the current beam, set the azimuth and elevation, and visualize the signal received by a static receiver on a 3D preview.

In this scenario, the "Frequency" is a read-only parameter, that displays the Frequency value of the "Current Emitter" (as selected in the "Emitter" dialog).

See:
- chapter 11.4, "How to Create and Configure Emitters", on page 186
- chapter 11.1, "Emitter Settings", on page 180
- chapter 6.2.1, "Common Scenario Settings", on page 55

Remote command:
`SCENario:EMITter` on page 364
`SCENario:EMITter:MODE` on page 365
`SCENario:EMITter:MODE:BEAM` on page 365
`SCENario:EMITter:CLEar` on page 282

Emitters (Collection) Scenario Settings
You can use "Emitter (Collection)" scenario test the receiver's ability to correctly detect the signal from different static emitters. In this dedicated scenario, you can configure several emitters and switch between them sequentially. You can also configure the receiver and change its position in the scanning beam of the emitters.



A scenario with several emitters requires a list of emitters. You can switch between these emitters and select which one is currently transmitting.

The "Current Emitter" parameter indicates the alias name of the currently selected one.

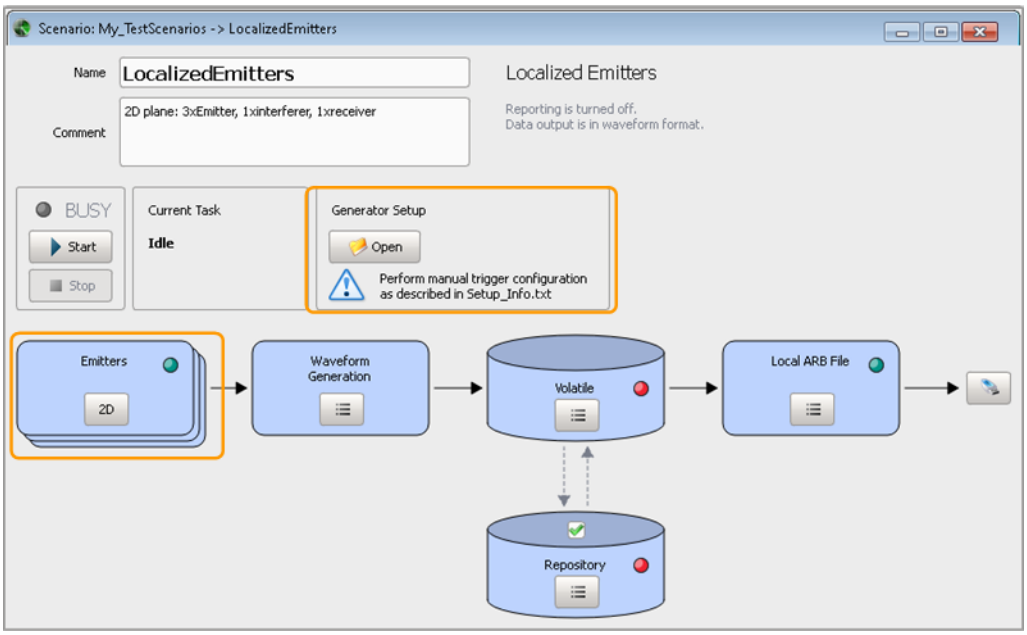
In this scenario, the "Frequency" is a read-only parameter, that displays the Frequency value of the "Current Emitter" (as selected in the "Emitter" dialog).

See:
- "To configure complex scenarios with several emitters" on page 189
- chapter 11.3, "Lists with Multiple Emitters", on page 185

- chapter 6.2.1, "Common Scenario Settings", on page 55

Remote command:
`SCENario:CEMit:CURRent` on page 363

Localized Emitters Scenario Settings



You can use "Localized Emitter" scenario to test the receiver's ability to correctly detect the sum signal from different static emitters and interferers. In this dedicated scenario, you configure the signal of one ore more emitters and optional one or more interferers that would be received by a receiver with defined characteristics. You can also configure the receiver and change its position in the scanning beam of the emitters.

Select "Emitters > 2D" to display the 2D view of the receiver and currently configured emitters and interferes, together with their main characteristics.

See:
- chapter 13.2, "How to Create Scenarios with Emitters, Interferers and a Receiver", on page 209
- chapter 19.2, "Signal to Generator Mapping Settings", on page 250
- chapter 6.2.1, "Common Scenario Settings", on page 55

When you create ARB files ("Loacl ARB File > Target > File"), select "Generator Setup > Open" to access the generated waveforms and the `Setup_Info.txt` file. Follow the instructions in this `Setup_Info.txt` file.

See also "To create, transfer and play the waveforms of 2D scenario manually" on page 258.

6.3 How to Select and Create a Test Scenario

The overview in table 6-1 helps you decide which scenario type is suitable for a certain test case.

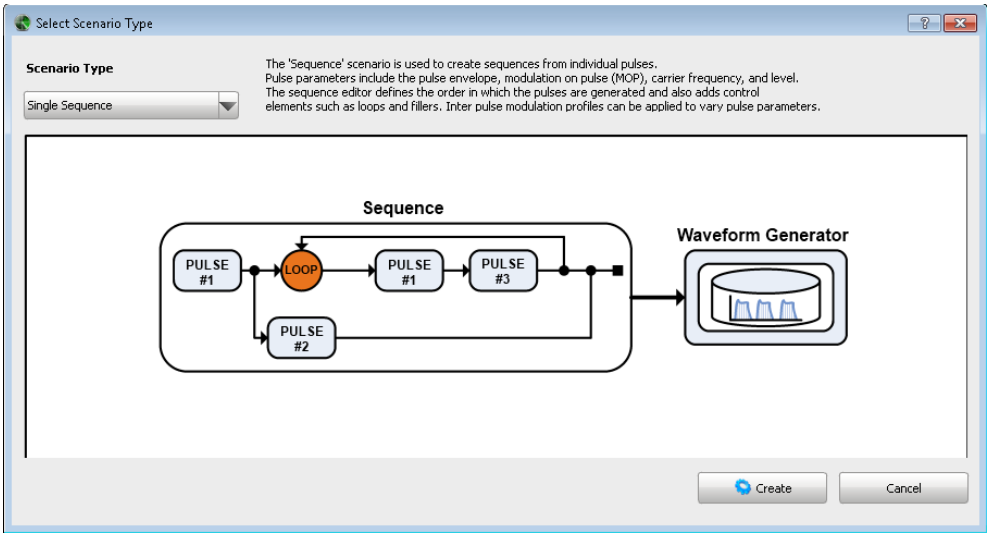
Table 6-1: Typical test cases with their suitable scenario types

Test case	Scenario type
Low level receiver tests Receiver tests with IPM profiles and modulation on pulse	Single Sequence Sequences (Collection) Waveform Sequence
Advanced receiver tests Receiver tests with antenna patterns and scans as well as reception of multiple signals at the same time	Single Emitter Emitters (Collection) Localized Emitters

To create a single pulse train scenario

1. To access the scenario settings dialog, perform one of the following:
   a) Select "Repository Tree > Scenario > New".
   b) In the "Toolbar", select the scenario icon .

   A scenario wizard opens that helps you to select a predefined scenario configuration.


Select Scenario Type
Scenario Type
Single Sequence
The 'Sequence' scenario is used to create sequences from individual pulses.
Pulse parameters include the pulse envelope, modulation on pulse (MOP), carrier frequency, and level.
The sequence editor defines the order in which the pulses are generated and also adds control elements such as loops and fillers. Inter pulse modulation profiles can be applied to vary pulse parameters.
Sequence
PULSE #1
LOOP
PULSE #1
PULSE #3
PULSE #2
Waveform Generator
Create
Cancel

2. Select "Scenario Type > Single Sequence".

   The dialog displays a short description of the current scenario.
   See also chapter 6.1, "Overview of the Available Scenarios and Their Complexity", on page 53.

3. Confirm with "Create".

   Created is a new scenario with default name `Scenario <n>`, where `n` is a number starting at one.
   The provided settings depend on the selected scenario type and are described in chapter 6.2, "Scenario Settings", on page 54.

4. Add information to describe and identify the scenario, like a name or a comment.

5. Use the standard functions in the context menus to select or create new sequences, select the signal generator, etc.

See table 3-4.

Perform further configurations, see:

- chapter 8.3, "How to Create Sequences and Use the Control Elements", on page 117
- chapter 14, "Creating Generator Profiles and Configuring the Instruments", on page 216
- chapter 18.3, "How to Analyze the Content of Waveform Files and Files with I/Q Data", on page 245

7 Creating a Pulse Library

This section provides background information on pulse generation and pulse characteristics, explains the provided settings and how to use them to create a pulse library.

- Basics on Pulse Signals and Pulse Generation 66
- Pulse Settings 68
- How to Create a New Pulse and Adjust Its Settings 98

7.1 Basics on Pulse Signals and Pulse Generation

Some background knowledge on basic terms and principles used in pulse generation is provided here for a better understanding of the required configuration settings.

The pulse parameters to be configured are based primarily on the IEEE 181 Standard 181-2003. For detailed descriptions refer to the standard documentation ("IEEE Standard on Transitions, Pulses, and Related Waveforms", from the IEEE Instrumentation and Measurement (I&M) Society, 7 July 2003).

Basic terms

The definition of the pulse parameters depends on the used timing profile (see Standard Timing Profile) and distinguishes between:

- **Power** 10/50/90
- **Voltage** 10/50/90
- (Power/Voltage) 10/100

Where the name of the timing profile follows the naming convention:

- Power/Voltage: indicates whether power or voltage based reference levels are used
- Indication of the used reference levels, i.e.:
  - 10/50/90 means that three reference levels are used: the lower (10% nominally), the mesial (50% nominally) and the upper (90% nominally)
  - 10/100 indicates that two levels are used: the base (0%) and the top (100%) level

A pulsed signal is a signal whose carrier power is modulated by two states: ON and OFF.

If the top power level of a pulse is not constant, this is called an amplitude **droop**.

The graphic on figure 7-1 illustrates the main pulse parameters and characteristic values when a **Power** 10/50/90 profile is used. If voltage profile is used, the definition of the pulse parameters is identical, but the parameters are related to **voltage**.

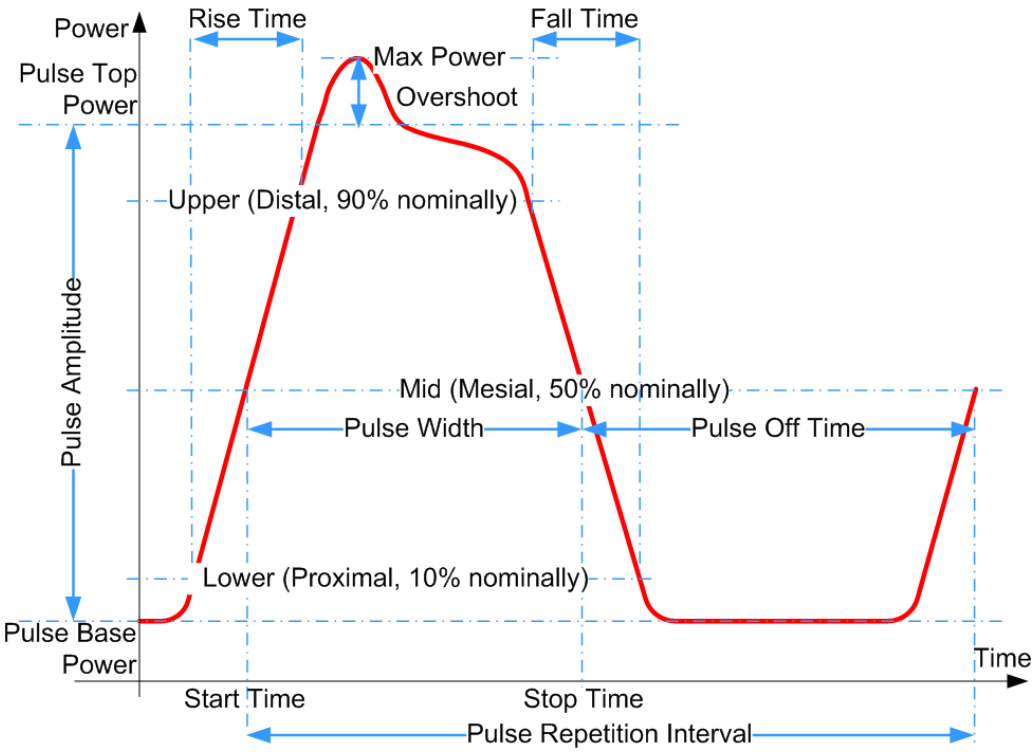


Fig. 7-1: Definition of the main pulse parameters and characteristic values ("Timing Profile > Power 10/50/90")

Pulse parameters ("Timing Profile > 10/50/90")

The following definitions apply:

- **Rise time, Fall time** (also known as transition duration of the first and the second transitions)
  That is the difference between two reference levels (10% and 90 % nominally) of the same transition.
- **Pulse width** (also known as pulse ON time or pulse duration)
  That is the difference between the first and the second transition of one pulse, i.e. the duration the signal remains above the 50% reference level.
- **Pulse Off time** (also known as pulse separation)
  The duration between the 50% reference level of the second transition of one pulse and the 50% reference level of the first transition of the subsequent pulse in the same pulse train.
- **Pulse repetition interval** (also known as pulse period)
  Is defined as the duration of one complete cycle consisting of:
  – the pulse width (i.e. pulse ON time or pulse duration)
  – the pulse Off time (pulse separation)

Pulse parameters ("Timing Profile > 0/100")

Note that the pulse parameters differ, if a "Timing Profile > 0/100" is used.

The following definitions apply:

- **Rise time, Fall time**
  That is the difference between the 0% and 100% reference levels of the same transition, i.e. between the base and the top level.
- **Pulse width** (pulse ON time)
  The duration the signal power remains at the top level (100% reference level).
- **Pulse Off time**
  The duration the signal power remains at the base level (0% reference level).
- **Pulse repetition interval**
  is defined as the duration of one complete cycle consisting of:
  – the rise time
  – the pulse width
  – the fall time
  – the pulse off time

For description of the related settings, see:

- chapter 7.2.3, "Pulse Level Settings", on page 73
- chapter 7.2.4, "Pulse Envelope Settings", on page 76

7.2 Pulse Settings

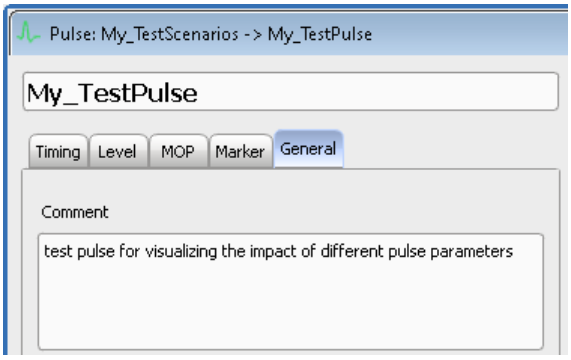
To access these settings:

► Select "Repository Tree > Pulse > New".

Pulses are characterized by the following settings:

- Pulse Name and Comment....69
- Pulse Timing Settings....70
- Pulse Level Settings....73
- Pulse Envelope Settings....76
- Modulation on Pulse (MOP) Settings....79
- Built-In Modulation Types and their Settings....82
- Envelope and Modulation Graphs....96

7.2.1 Pulse Name and Comment



Pulse Name....69
Comment....69

Pulse Name
Displays the name of the current pulse.

Remote command:
`PULSe:CREate` on page 277
`PULSe:NAME` on page 278
`PULSe:CATalog?` on page 276
`PULSe:SELect` on page 277
`PULSe:REMove` on page 278

Comment
General pulse information like comment, etc.

Remote command:
`PULSe:COMMent` on page 278

7.2.2 Pulse Timing Settings

To access these settings:

► Select "Pulse > Timing".

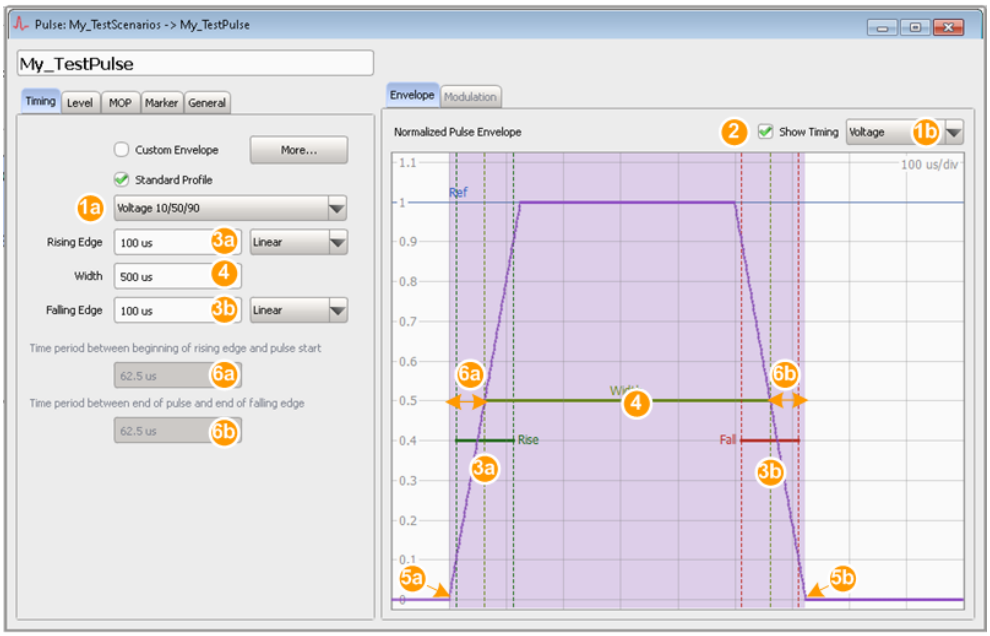


Fig. 7-2: Pulse Timing: understanding the displayed information

1a = "Standard Profile > Voltage 10/50/90"
1b = "Pulse Envelope Units > Voltage"
2 = "Show Timing > On" indicates the pulse timing parameters on the envelope graph
3a = "Rising Edge = 100 us", i.e. the time it takes the voltage to rise from 10% to 90% of the top level
3b = "Falling Edge = 100 us", i.e. the time it takes the voltage to fall from 90% to 10% of the top level
4 = "Width = 500 us", i.e. the duration the voltage is above 50 % of the top level
5a, 5b = Beginning of the rising and falling edge
6a = "Time period between beginning of the rising edge and pulse start = 62.5 us", i.e. the time it takes the voltage to rise from 0% to 50% of the top level
6b = "Time period between end of pulse and the end of falling edge = 62.5 us", i.e. the time it takes the voltage to fall from 50% to 0% of the top level

Use the following parameters to define the pulse in the time domain:

Custom Envelope....70
Standard Timing Profile....71
Rising/Falling Edge, Width, Rising/Falling Slope....71
Time period between beginning of rising edge and pulse start/Time period between end of pulse and end of falling edge....72

Custom Envelope
Applies an envelope shaping according to a custom envelope function.

See chapter 7.2.4, "Pulse Envelope Settings", on page 76.

Remote command:
`PULSe:CUSTom` on page 335
`PULSe:ENVelope:MODE` on page 337

Standard Timing Profile
Enables and selects a predefined standard envelope profile.

Select from a voltage-based "Voltage 10/50/90", a power-based "Power 10/50/90" or a 0-100% "0/100" timing profile.

For brief introduction to the timing profiles and their impact on the pulse timing, see chapter 7.1, "Basics on Pulse Signals and Pulse Generation", on page 66.

See figure 7-3 for illustration of the pulse width, pulse rise and fall time, and period between the beginning of the rising edge and the pulse start for each of the envelope profiles.

Enable "Show Timing" to visualize the timing parameters. Note that the "Envelope Graphs" use different "Normalized Pulse Envelope" settings.

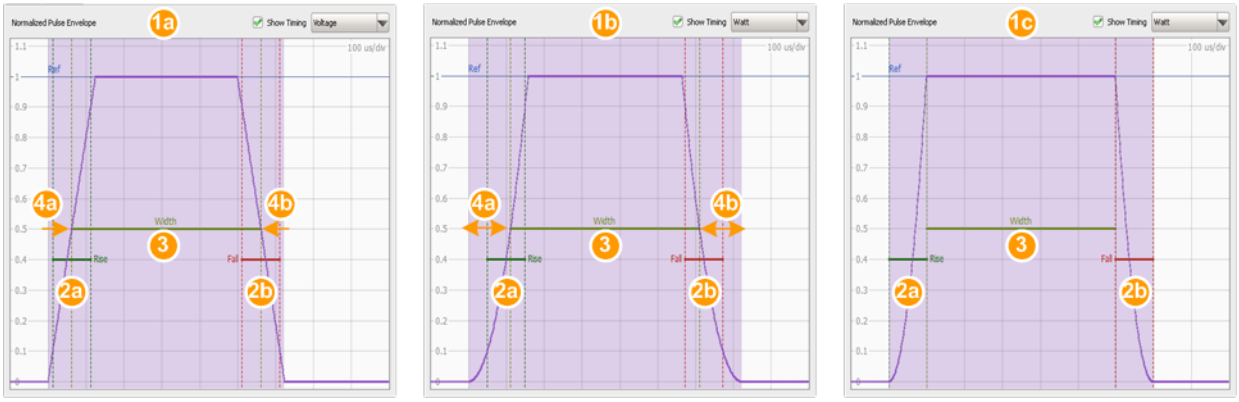


Fig. 7-3: Pulse timing parameters and the envelope profiles (linear rising and falling slopes)

1a = "Voltage 10/50/90" profile
1b = "Power 10/50/90" profile
1c = "0/100" profile
2a = "Rising Edge = 100 us"
2b = "Falling Edge = 100 us"
3 = "Width = 500 us"
4a = "Time period between beginning of the rising edge and pulse start" (with"0/100" profile this time is 0 us)
4b = "Time period between end of pulse and the end of falling edge" (with"0/100" profile this time is 0 us)

Remote command:
`PULSe:TIME:REFerence` on page 348

Rising/Falling Edge, Width, Rising/Falling Slope
These parameters define the pulse shape.

The current pulse shape is displayed on the Envelope Graph. To visualize the timing parameters on the graph, select "Envelope > Show Timing > On".

The parameter definition depends on the selected Standard Timing Profile, see:
- "Pulse parameters ("Timing Profile > 10/50/90")" on page 67
- "Pulse parameters ("Timing Profile > 0/100")" on page 67

The total pulse duration and the pulse off time are calculated automatically from the selected pulse width, rise and fall time, and the PRI (see PRI).

"Rising/Falling Edge"
Transition time of the rising/falling edge.

Remote command:
`PULSe:TIME:RISE` on page 348
`PULSe:TIME:FALL` on page 348

"Rising Slope Type"
Form of the rising/falling slope.
Available are linear, cosine, raised and root cosine. You can also use arbitrary envelope data or import you custom shapes, see chapter 16, "Defining Complex Modulation Schemes and IPM Profiles", on page 232.

Remote command:
`PULSe:TYPE:RISE` on page 349
`PULSe:TYPE:FALL` on page 349

"Width" Pulse duration.

Remote command:
`PULSe:TIME:WIDTh` on page 349

Time period between beginning of rising edge and pulse start/Time period between end of pulse and end of falling edge
Display information on the time periods during that the RF power is attenuated or totally suppressed but per default there is modulation or data content present.

Note: Avoiding data truncation.
To avoid truncation of data during the rising time, add guard bits to the data source.

Per default, modulation is already present during the rising and falling phases. You can restrict the pulse area on that the modulation is applied, see "Restrict modulation to a certain area of the pulse" on page 80.

7.2.3 Pulse Level Settings

To access these settings:

► Select "Pulse > Level".

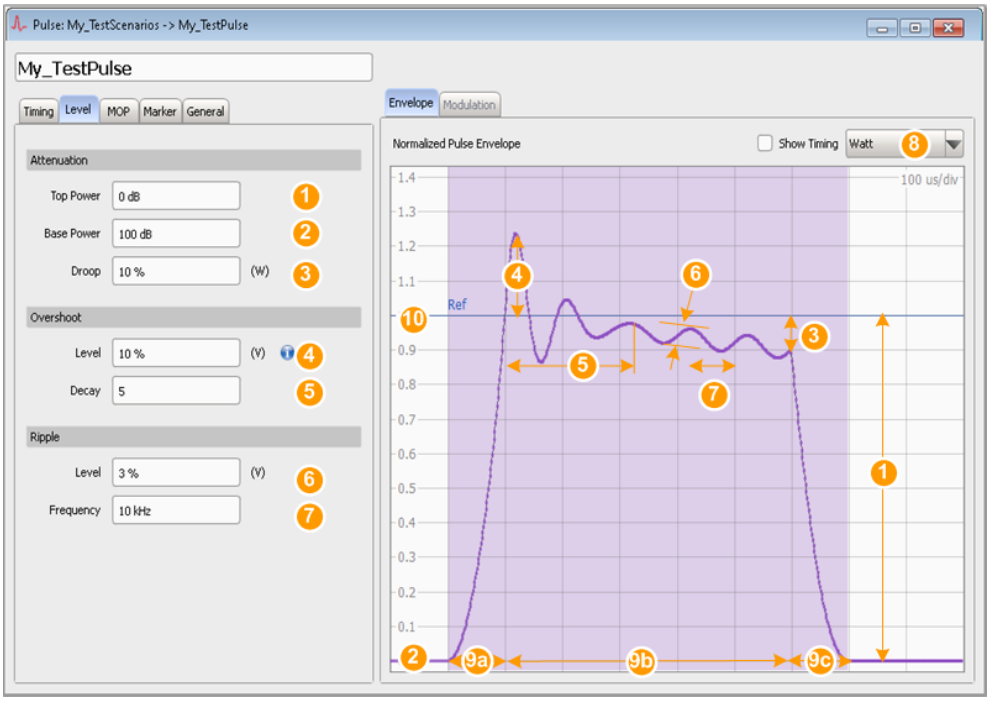


Fig. 7-4: Pulse Level: understanding the displayed information

1 = Pulse top power corresponds to 1000mW ("Pulse Envelope = Watt"), i.e. little or no attenuation ("Attenuation Top Power = 0 dB")
2 = "Base Power", i.e. high attenuation; typically 100 dB
3 = Amplitude change from the beginning to the end of the pulse ON time
4 = Overshoot level, in Volt (diagram uses Watts)
5 = Damping constant that influences the time it takes the overshoot to decay
6 = Amplitude of the oscillating signal, in Volt (diagram uses Watts)
7 = Signal oscillating with the selected frequency ("Frequency = 10 kHz", i.e. the oscillation period is 100 us)
8 = Pulse amplitude units (Y-axis units)
9a, 9b, 9c = Pulse timing: "Profile > Standard 0/100", "Rising Edge = Falling Edge = 100 us", "Width = 500 us" (see Standard Timing Profile)
10 = Reference value used by the calculation of the pulse level parameters: "Ref. = 0 dB (1 Watt)"; the value is set with the parameter Ref. Level

Use the following parameters to define the power/amplitude pulse parameters:

Attenuation....73
Overshoot....74
Ripple....75

Attenuation
Simulates changes in the pulse amplitude.

"Top Power" The pulse top power is used as a reference (100%) to determine other parameter values such as the rising/falling thresholds.
Typically, the "Top Power" uses a little or no attenuation, such as 0 dB.
See also figure 7-1.

"Base Power" The pulse base power is used as a reference (0%) to determine other parameter values such as the rising/falling thresholds.
Typically, the "Base Power" uses a high attenuation, such as 100 dB.
You can change this value if you use a "0/100" profile.
The difference between the "Top Power" and the "Base Power" is the pulse amplitude.

"Droop" Droop is a measure for the amplitude change from the beginning to the end of the pulse.
The percentage ratio values are calculated in %W. A positive value decreases the amplitude.

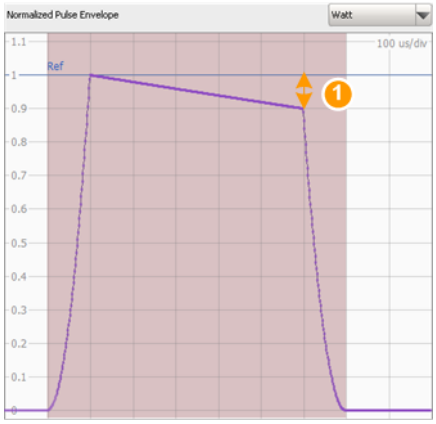


Fig. 7-5: Illustration of the effect of the amplitude droop (Profile = 0/100)

1 = "Droop = 10% (W)"

Remote command:
`PULSe:LEVel:ON` on page 337
`PULSe:LEVel:OFF` on page 337
`PULSe:LEVel:DROop` on page 337

Overshoot
Simulates that the signal exceeds the pulse's "Top Power".

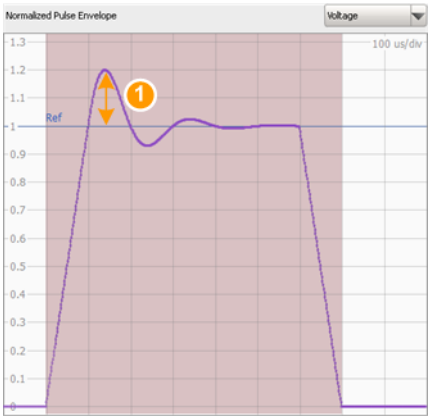


Fig. 7-6: Illustration of the effect of the overshoot parameters (Profile = 0/100)

1 = "Overshoot = 20%" (value in percentage related to Volts)

"Level" The height of the local maximum (Max. Power) after a rising edge, divided by the pulse amplitude, see figure 7-1.

"Decay" Sets the normalized decay constant. It influences the time it takes the overshoot to decay.
"Decay = 1" is the aperiodic border case. The bigger the value the longer the decay time.

Remote command:
`PULSe:OVERshoot` on page 347
`PULSe:OVERshoot:DECay` on page 347

Ripple
Simulates the unwanted ringing effect, i.e. that the signal oscillates around the power level. This oscillation is defined with a frequency and level.

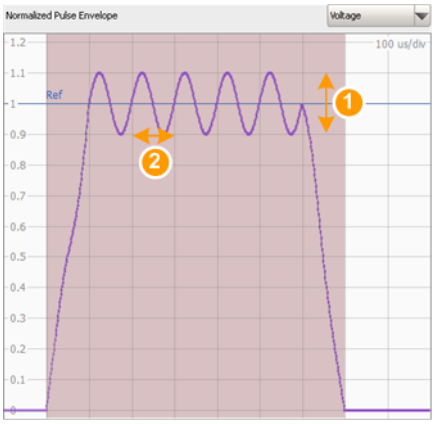


Fig. 7-7: Illustration of the effect of the ripple parameters (Profile = 0/100)

1 = "Level = 20% " (value in percentage related to Volts)
2 = "Frequency = 10 kHz", i.e. the oscillation period is 100 us

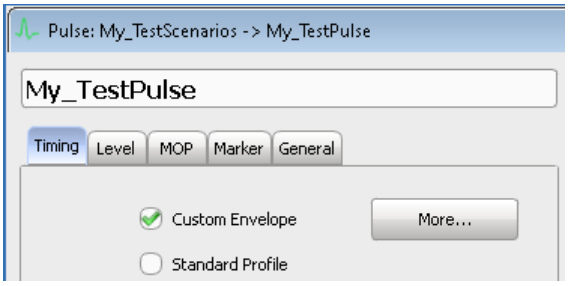
"Level" The ripple is calculated as the difference between the maximum and minimum deviation from the pulse top reference, within a user specified interval.
The percentage ratio values are calculated in %W if the "Pulse Envelope" is defined in W, otherwise in %V.

"Frequency" Sets with which frequency the ripple oscillates.

Remote command:
`PULSe:RIPPle` on page 348
`PULSe:RIPPle:FREQuency` on page 348

7.2.4 Pulse Envelope Settings

Instead of defining a pulse by its rise, on and fall time you can use your custom envelope shape.



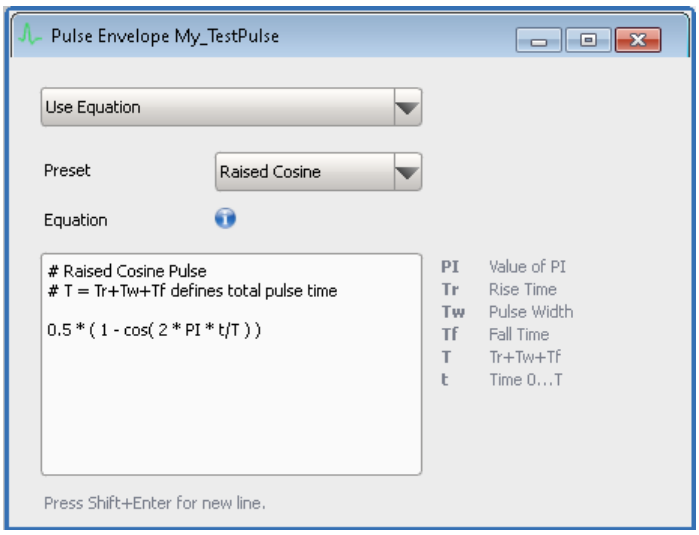
Custom envelopes are described in one of the following ways:

- As an equation, see "Envelope Definition as an Equation" on page 76
- As a function of imported I/Q data, see "Envelope Definition as a Function form Imported Data" on page 77

Both ways affect the level versus time and can be used with any kind of modulation (MOP). Custom envelopes are multiplied with the existing pulse shape as defined with the timing parameters. The software maps the custom envelope to the pulse shape.

Envelope Definition as an Equation..... 76
Envelope Definition as a Function form Imported Data.....77

Envelope Definition as an Equation
Envelope is described by an equation, as selected with the parameter "Preset".



The software provides a selection of envelope shapes. Upon selection, the dialog displays the used equation as well as information on the used parameters and variables.

For description of the provided envelope functions, see "Equations and parameters used to define custom pulse envelopes" on page 399.

See also:

- "To generate a pulse with raised cosine envelope shape" on page 103
- "To define your custom envelope shape with an equation" on page 104

Remote command:

`PULSe:ENVelope:EQUation` on page 336

Envelope Definition as a Function form Imported Data

You can define an envelope function, store it in a file, and import and load files describing envelope functions.

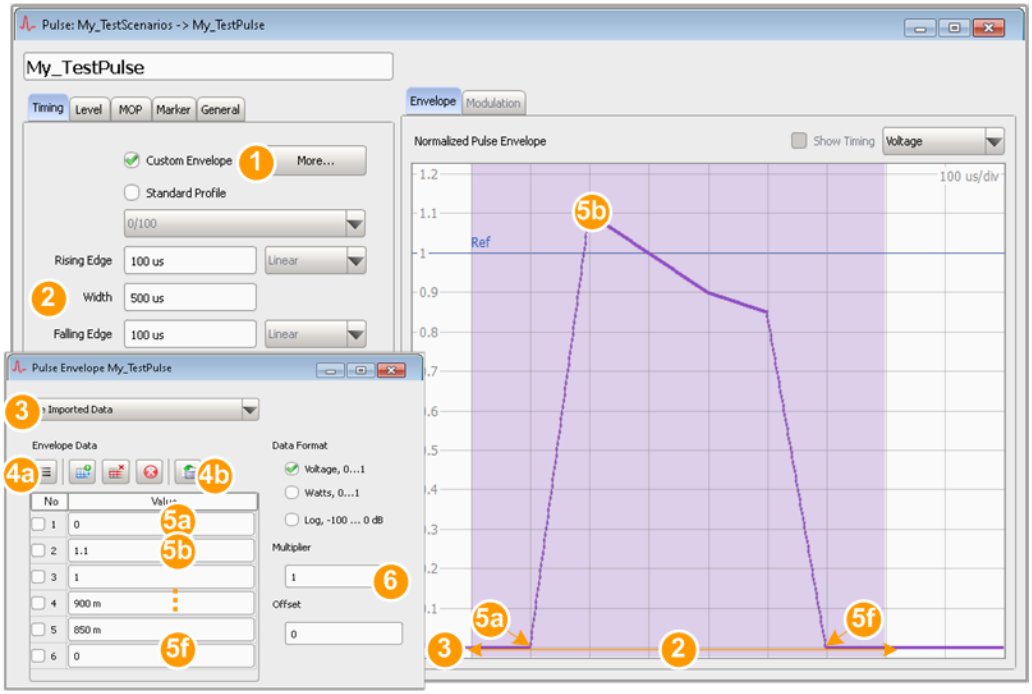


Fig. 7-8: "Custom Envelope > Use Imported Data": understanding the displayed information

1 = "Custom Envelope"
2 = The duration between the beginning of the rising edge and the end of the falling edge is 100 us + 500 us + 100 us= 700 us; i.e. the x-values of the 6 items are distributed equidistant, at each 100 us
3 = "Use Imported Data"
4a = Icons that access standard functions
4b = "Import Data from File" access a dialog for loading of custom envelopes
5a, 5b, ... 5e = "Value", i.e. value on the y-axis
6 = "Offset = 0"

"Envelope Data Table"
An envelope curve consists of two or more items. Each item is a line, defined by the y-value of end point; the start point is allays the end point of the previous item.
The X-values are calculated automatically so that the items are distributed equidistant between the beginning of the rising edge an the end of the falling edge.

"Data Format" Selects the scaling used by the Envelope Graph.

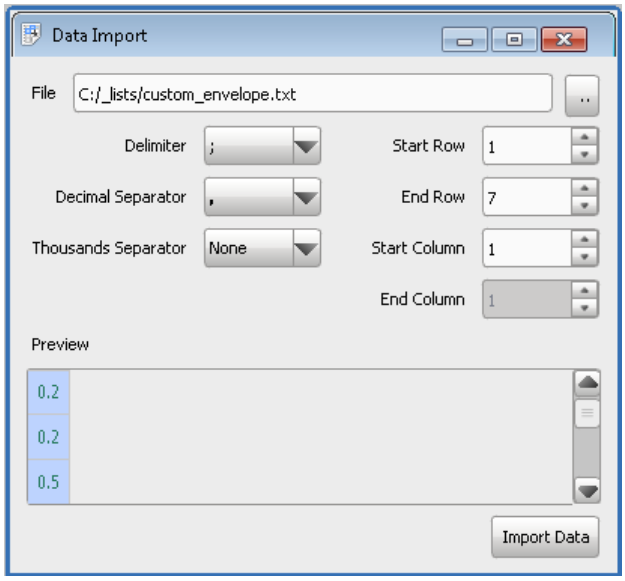
"Multiplier" Applies a multiplier to the Y-values.

"Offset" Offsets the envelope.

 Icons that access standard functions, like add or delete row, etc. (see also table 3-4)
Use the functions in the context-sensitive menu to append new item or reorder the existing item.

"Import Data from File"

Standard file management function to load custom envelopes from and as a ASCII file with `*.txt` or `*.csv` file extension.
When file is loaded, the "Data Import" dialog assists you to select the used delimiter and separator to correctly interpret the file content.



The provided settings are self-explanatory.

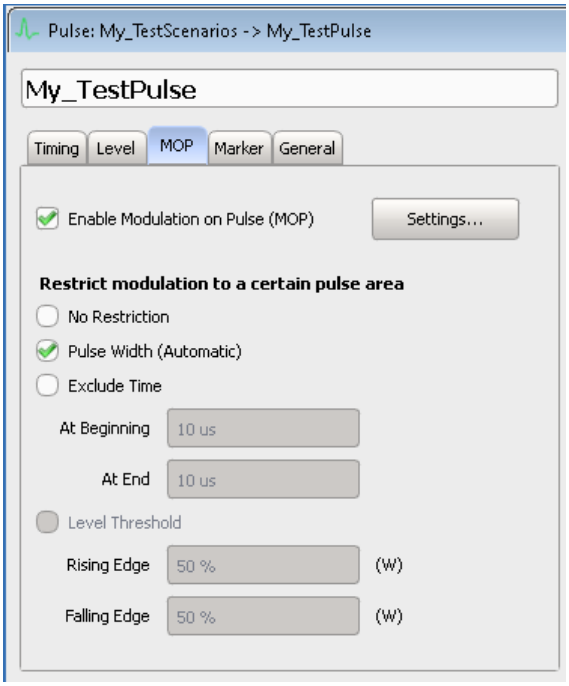
Remote command:
`PULSe:ENVelope:DATA:ITEM:ADD` on page 279
`PULSe:ENVelope:DATA:ITEM:SELect` on page 280
`PULSe:ENVelope:DATA:ITEM:DELete` on page 281
`PULSe:ENVelope:DATA:ITEM:VALue` on page 335
`PULSe:ENVelope:DATA:UNIT` on page 335
`PULSe:ENVelope:DATA:MULTiplier` on page 336
`PULSe:ENVelope:DATA:OFFSet` on page 336
`PULSe:ENVelope:DATA:LOAD` on page 335
`PULSe:ENVelope:DATA:CLEar` on page 282

7.2.5 Modulation on Pulse (MOP) Settings

To access these settings:

1. Select "Pulse > MOP".

2. Select "Enable Modulation on Pulse (MOP)".



Use the following common parameters to define and enable modulation on the pulses:

Enable Modulation on Pulse (MOP)....80
Settings....80
Restrict modulation to a certain area of the pulse....80

Enable Modulation on Pulse (MOP)
Enables/disables that the pulse signal is modulated. If the MOP is disabled, the software generates a envelope signal based on the pulse level and timing parameters.

Remote command:
`PULSe:MOP:ENABle` on page 342

Settings...
Access the dialog with modulation settings, see Built-In Modulation Types and their Settings.

Restrict modulation to a certain area of the pulse
Customizes (restricts) the pulse area on that the MOP will be applied.

With its default settings ("Configure > Colors > Preset > Classic"), the software indicates (highlights) the pulse area on that the MOP is applied. See figure 7-9 and compare the colors used to indicate the pulse (lilac) and the pulse area on that MOP is applied (pastel orange).

See also chapter 3.7.2, "Changing Colors and Default Configuration", on page 36.

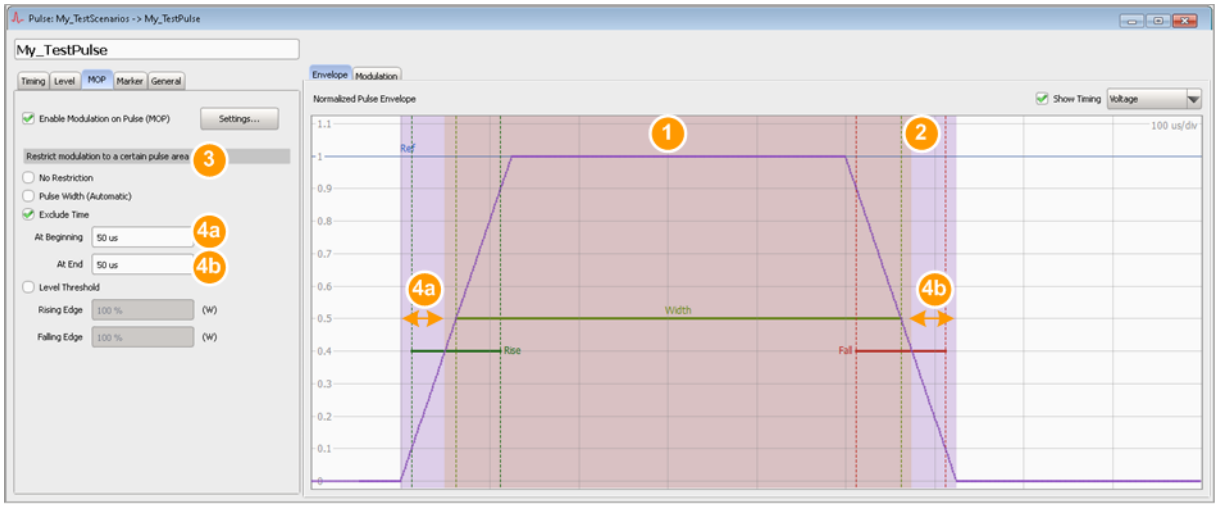


Fig. 7-9: Restrict modulation to a certain area of the pulse: understanding the displayed information ("Timing Profile > Voltage 10/50/90")

1 = Orange color: indicates the pulse area on that the MOP is applied
2 = Lilac color: indicates the pulse
3 = "Restrict modulation to a certain area of the pulse > Exclude Time"
4a, 4b = Excluded are 50 us from the beginning of the rising edge and at the end of the falling edge; note that the pulse are the MOP is applied on is wider than the pulse width

"No restrictions"
The MOP is applied form the beginning of the rising edge until the end of the falling edge.

"Pulse Width (Automatic)"
The MOP is applied during the time specified as pulse width time ("Pulse > Timing > Width").

"Exclude time" Defines a time span to be excluded at the beginning of the rising edge and at the end of the falling edge.

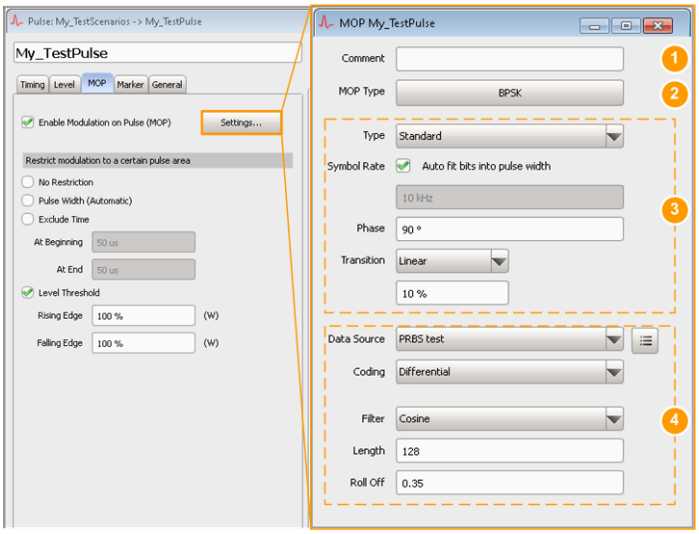
"Level Threshold"
The MOP is applied during the time the signal level is greater than the specified level limits.

Remote command:
`PULSe:MOP:EXCLude:ENABle` on page 342
`PULSe:MOP:EXCLude:MODE` on page 343
`PULSe:MOP:EXCLude:TIME:STARt` on page 343
`PULSe:MOP:EXCLude:TIME:STOP` on page 343
`PULSe:MOP:EXCLude:LEVel:STARt` on page 342
`PULSe:MOP:EXCLude:LEVel:STOP` on page 342

7.2.6 Built-In Modulation Types and their Settings

To access these settings:

► Select "Pulse > MOP > Settings".



1 = "Comment"
2 = "MOP Type" select a modulation scheme
3 = Modulation parameters, depending on the selected "MOP Type"
4 = "Data source, Coding and Filter" settings (if required by the selected "MOP Type")

Use the parameter "MOP Type" to select a modulation scheme and define the respective modulation parameters. You can also add a description in the "Comment" field.
Remote commands:
`PULSe:MOP:COMMent` on page 278
`PULSe:MOP:TYPE` on page 347
Further available settings depend on the selected modulation.

- AM and ASK Modulation
- FM and FSK Modulation
- Chirp Modulation
- Phase Modulation
- QPSK Modulation
- Noise
- Plugins

The Modulation Graph displays the characteristics of the currently used MOP.

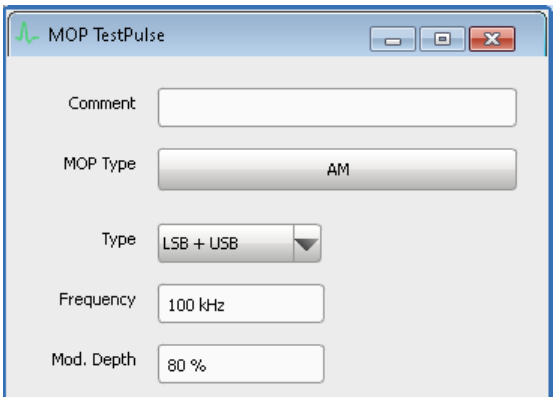
Some MOP types require a data source. For description of the related settings, see chapter 7.2.6.8, "Data Source, Coding, Filter", on page 95.

7.2.6.1 AM and ASK Modulation

The amplitude modulation (AM) and amplitude shift key modulation (ASK) are modulation schemes that vary the amplitude of the signal.

AM....83
ASK....83
AM Step....84

AM
Enables an AM with a single tone.



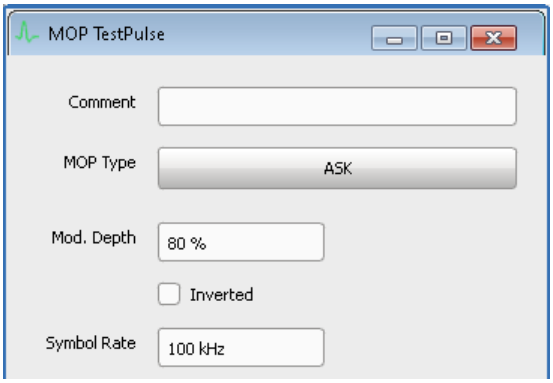
"AM Type" Modulation method, like a double-sideband ("Standard"), a Low Side Band ("LSB"), a Upper Side Band ("USB"), or a double sideband with suppressed carrier ("LSB+USB") amplitude modulation.

"Frequency" Modulation frequency.

"Modulation Depth"
Depth of the modulation signal in percent, i.e. the peak change in the RF amplitude from its unmodulated value to the amplitude of the unmodulated carrier.

Remote command:
`PULSe:MOP:AM:TYPE` on page 338
`PULSe:MOP:AM:FREQuency` on page 337
`PULSe:MOP:AM:MDEPth` on page 337

ASK
Enables an ASK where the amplitude of the RF carrier is attenuated for a symbol value of zero and remains at full level for symbol values of one. The level of attenuation is specified as depth in percent.



"Modulation Depth"
Depth of the modulation signal in percent.
$m = (Amplitude_{max}-Amplitude_{min}) / (Amplitude_{max}+Amplitude_{min})$

"Inverted" Inverts the amplitude-shift keying (ASK) modulation.

"Symbol Rate" Symbol rate of the modulating signal.

"Data Source, Coding, Filter"
See:

- chapter 15.1, "Data Sources Settings", on page 228
- "Coding" on page 95
- "Filter" on page 95

Remote command:
`PULSe:MOP:ASK:MDEPth` on page 339
`PULSe:MOP:ASK:INVert` on page 339
`PULSe:MOP:ASK:SRATe` on page 339

AM Step
Defines a modulation as a number of discrete states (table rows), each described with its "Duration" and "Level".

Use the standard "Append", "Remove Last", or "Delete All" functions to add or remove a row. The provided settings are self-explanatory.

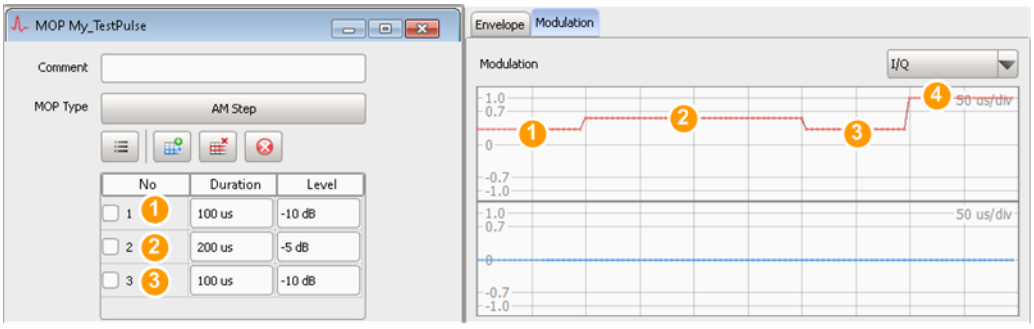


Fig. 7-10: "MOP Type > AM Step": understanding the displayed information ("Pulse Width = 500 us", "Restrict MOP to > Pulse Width")

1, 2, 3 = Three discrete states with different attenuation values and a total duration of 400 us
4 = The last 100 us are not attenuated

If the total time of the defined states is shorter than the duration of the pulse area on that the MOP should be applied, an attenuation of 0 dB is assumed for the remaining time.

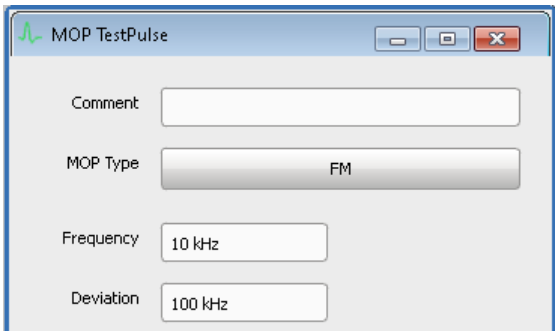
Remote command:
`PULSe:MOP:AMSTep:ADD` on page 279
`PULSe:MOP:AMSTep:COUNt?` on page 280
`PULSe:MOP:AMSTep:SELect` on page 280
`PULSe:MOP:AMSTep:INSert` on page 281
`PULSe:MOP:AMSTep:LEVel` on page 338
`PULSe:MOP:AMSTep:DURation` on page 338
`PULSe:MOP:AMSTep:CLEar` on page 282
`PULSe:MOP:AMSTep:DELete` on page 281

7.2.6.2 FM and FSK Modulation

The frequency modulation (FM) and frequency shift keying (FSK) modulation are modulation schemes that vary the frequency of the signal.

FM 85
FSK 85
FM Step 86

FM
Enables a FM with a single tone.

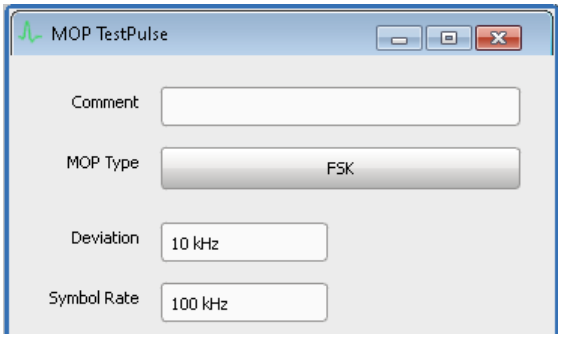


"Frequency" Modulation frequency

"Deviation" FM frequency deviation

Remote command:
`PULSe:MOP:FM:FREQuency` on page 345
`PULSe:MOP:FM:DEViation` on page 344

FSK
Enables a FSK. High symbols set the frequency to "+Deviation" and low symbols to "–Deviation".



"Deviation" FSK frequency deviation

"Symbol Rate" Symbol rate of the modulating signal.

"Data Source, Coding, Filter"
See:

- chapter 15.1, "Data Sources Settings", on page 228
- "Coding" on page 95
- "Filter" on page 95

Remote command:
`PULSe:MOP:FSK:DEViation` on page 345
`PULSe:MOP:FSK:SRATe` on page 345

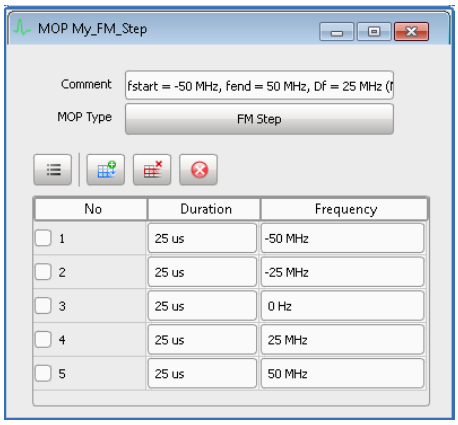
FM Step
The FM step modulation is a common pulse compression technique. It defines a modulation as a number of discrete states (table rows), each described with its "Duration" and "Frequency".

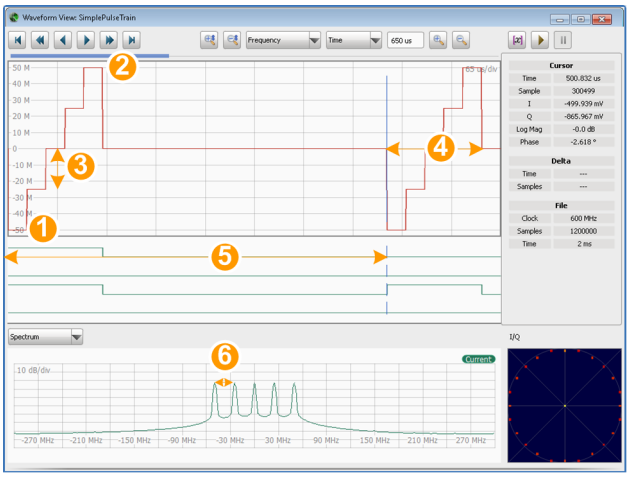
Use the standard "Append", "Remove Last", or "Delete All" functions to add or remove a row. The provided settings are self-explanatory.

Example: Sequence composed of four pulses with MOP=FM Step

Configure a FM Step with f_{start} = -50 MHz, f_{end} = 50 MHz, Step = 25 MHz, #Steps = 5.

Build a sequence and use the "Waveform View" dialog to visualize the signal, see chapter 18.3, "How to Analyze the Content of Waveform Files and Files with I/Q Data", on page 245.





Example: Sequence composed of four pulses with MOP=FM Step

- 1, 2: Start and stop frequency
- 3: Step = 25 MHz
- 4: Pulse width = 125 us
- 5: PRI = 500 us
- 6: Frequency spacing = 25 MHz

Remote command:

`PULSe:MOP:FMSTep:ADD` on page 279
`PULSe:MOP:FMSTep:COUNt?` on page 280
`PULSe:MOP:FMSTep:SELect` on page 280
`PULSe:MOP:FMSTep:INSert` on page 281
`PULSe:MOP:FMSTep:FREQuency` on page 338
`PULSe:MOP:FMSTep:DURation` on page 338
`PULSe:MOP:FMSTep:CLEar` on page 282
`PULSe:MOP:FMSTep:DELete` on page 281

7.2.6.3 Chirp Modulation

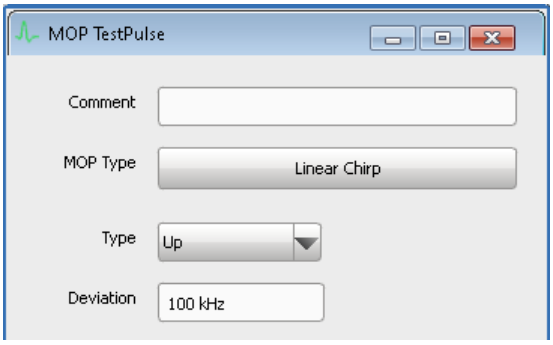
A chirp is a signal in which the frequency varies over the time. The chirp sweeps the RF signal across a set frequency range.

The chirp modulation scheme is used for pulse compression.

Linear Chirp....87
Equation....88
Polynomial Chirp....88

Linear Chirp

The linear chirp varies the frequency linearly over time.



"Type" Defines the frequency variation:

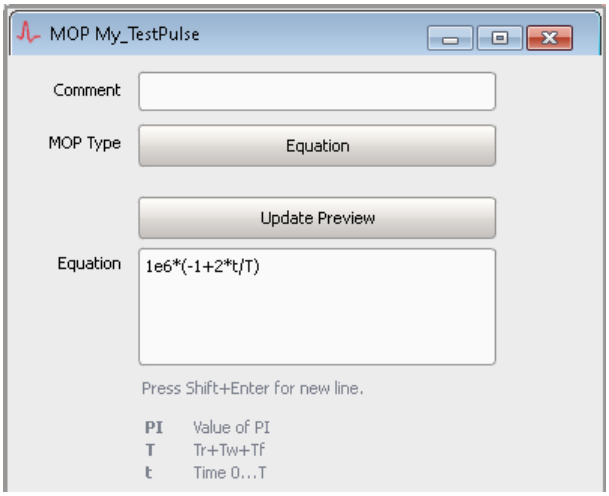
- Linearly increasing (UP)
- Linearly decreasing (Down)
- Following a full sine wave (Sine)
- Ascending and then descending (triangle)

"Deviation" Chirp deviation.

Remote command:
`PULSe:MOP:CHIRp:TYPE` on page 341
`PULSe:MOP:CHIRp:DEViation` on page 341

Equation
Define your custom chirp mathematically as a function of time, e.g. *1e6*(-1+2*t/T)*.

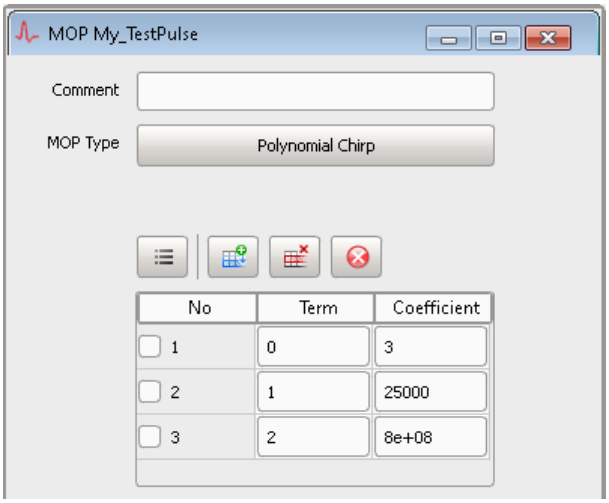
Select "Update Preview" to visualize the signal.



See also chapter A.2, "Formula Syntax", on page 398.

Remote command:
`PULSe:MOP:NLCHirp:EQUation` on page 345

Polynomial Chirp
Define the chirp as a polynomial with one or more "Term"/"Coefficient" value pairs.



The instantaneous frequency versus time is calculated according to the equation:

$f(t) = s^*\sum (a_0 + a_n{*}t^n)$, where:

- n = 1 .. 5
- the multiplier *s* and the coefficients a_n are values within the range – 10^6 to 10^6.

Remote command:
`PULSe:MOP:PCHirp:ADD` on page 279
`PULSe:MOP:PCHirp:COUNt?` on page 280
`PULSe:MOP:PCHirp:SELect` on page 280
`PULSe:MOP:PCHirp:INSert` on page 281
`PULSe:MOP:PCHirp:TERM` on page 346
`PULSe:MOP:PCHirp:COEFficient` on page 346
`PULSe:MOP:PCHirp:CLEar` on page 282
`PULSe:MOP:PCHirp:DELete` on page 281

7.2.6.4 Phase Modulation

Provided are the following phase modulation technics:

Barker....89
Poly Phase....90
Custom Phase....90
BPSK....91

Barker
Phase modulation with Barker codes results in signals with low autocorrelation properties.

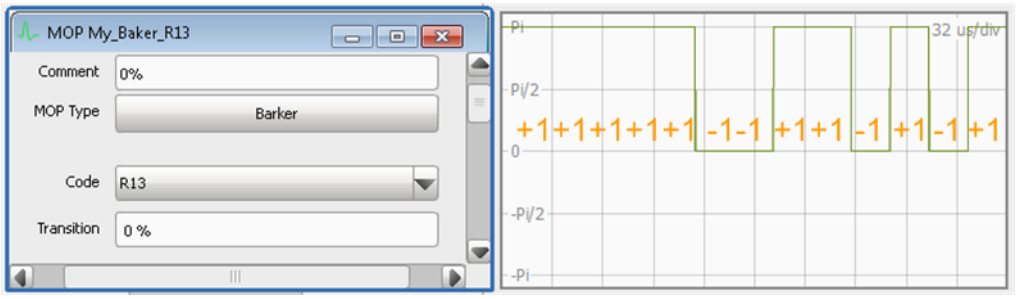


Fig. 7-11: "MOP Type > Barker": understanding the displayed information

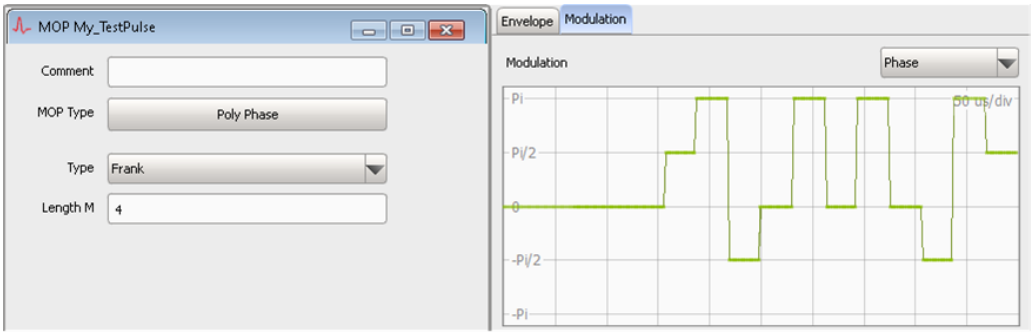
Barker codes are used for pulse compression and are defined by:

"Barker Code" A Barker code is a finite sequence of N values of +1 and −1; R13 for example is the '+1+1+1+1-1-1+1+1-1+1-1+1' sequence.

"Transition" Sets the transition duration for constant envelope operation.
A "Transition = 0%" is the border case, where the change for +1 to -1 or vise versa is performed immediately.
Observe the effect of a "Transition > 0%" on the displayed spectrum and on the I/Q constellation diagram.

Remote command:
`PULSe:MOP:BARKer:CODE` on page 339
`PULSe:MOP:BARKer:TTIMe` on page 340

Poly Phase
A polyphase code modulation is suitable for pulse compression and is mainly used in Low Probability of Intercept (LPI) radars.



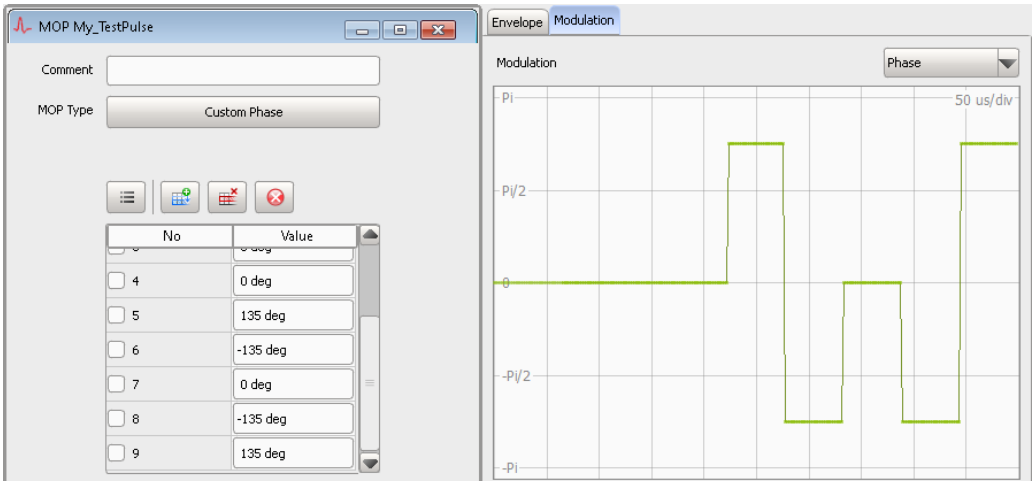
"Type" Selection of polyphase compression codes: the default Frank code, and four modified versions of the Frank code, the P1, P2, P3 and P4 codes.

"Length M" Sets the code order, i.e. the number of elements.

Remote command:
`PULSe:MOP:POLY:TYPE` on page 346
`PULSe:MOP:POLY:LENGth` on page 346

Custom Phase
Defines a sequence of phase values. The phase changes are distributed evenly.

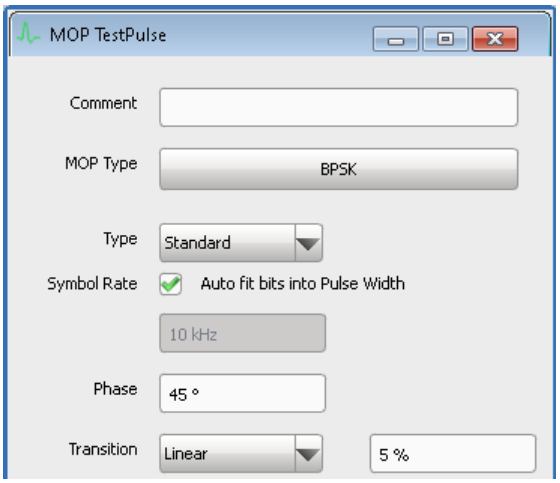
Use the standard "Append", "Remove Last", or "Delete All" functions to add or remove a row. The provided settings are self-explanatory.



Remote command:
PULSe:MOP:PLISt:ADD on page 279
PULSe:MOP:PLISt:COUNt? on page 280
PULSe:MOP:PLISt:SELect on page 280
PULSe:MOP:PLISt:INSert on page 281
PULSe:MOP:PLISt:VALue on page 346
PULSe:MOP:PLISt:CLEar on page 282
PULSe:MOP:PLISt:DELete on page 281

BPSK

A Binary Phase Shift Keying (BPSK) modulation applies a phase shift with a selected offset ("Phase") at each constellation point.



"Type" Selects the BPSK modulation method, like a standard BPSK modulation or a BPSK modulation with a constant envelope (C-BPSK).

"Symbol Rate" Symbol rate of the modulating signal, user defined or automatically estimated.

"Phase" Phase shift between the constellation point.

"Transition/Transition Function"

These parameters are particulary important, when a BPSK modulation with a constant envelope is used.
A "Transition = 0%" is a border case, where the change between two constellation points is performed immediately, i.e. the transition skips (switches back and forth on a direct line) between the constallation points.
The greather the value, the slower the transition. With a "Transition > 0%", the two constallation points are still exactly opposite, but the transition follows a circular arc.

The "Transition Function" defines how the values are distributed on the transition curve.

- "Linear": all values are uniformly distributed in the transition
- "Cosine": there are fewer values in the middle of the transition and accumulated values near to the two end constellation points.

"Data Source, Coding, Filter"

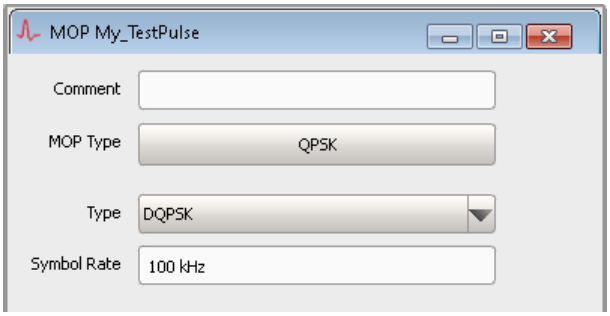
See:

- chapter 15.1, "Data Sources Settings", on page 228
- "Coding" on page 95
- "Filter" on page 95

Remote command:
`PULSe:MOP:BPSK:TYPE` on page 341
`PULSe:MOP:BPSK:SRATe` on page 340
`PULSe:MOP:BPSK:SRATe:AUTO` on page 340
`PULSe:MOP:BPSK:PHASe` on page 340
`PULSe:MOP:BPSK:TTYPe` on page 341
`PULSe:MOP:BPSK:TTIMe` on page 341

7.2.6.5 QPSK Modulation

A quadrature phase shift keying (QPSK) modulation applies a phase shift at each constellation point.



QPSK Type
Selects the QPSK modulation method, like a regular QPSK, an Offset QPSK ("OQPSK") or a Differential QPSK DQPSK modulations.

Remote command:
`PULSe:MOP:QPSK:TYPE` on page 347

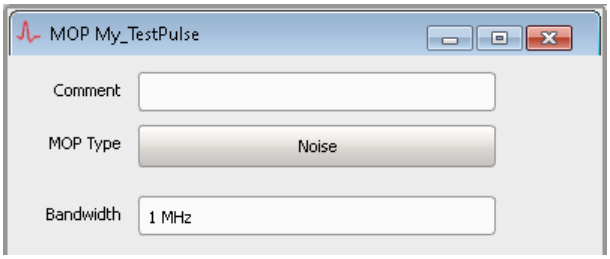
Symbol Rate
Symbol rate of the modulating signal.

Remote command:
`PULSe:MOP:QPSK:SRATe` on page 347

7.2.6.6 Noise

Generates an AWGN noise signal with selected "Bandwidth".

The noise is superimposed during all pulse phases.



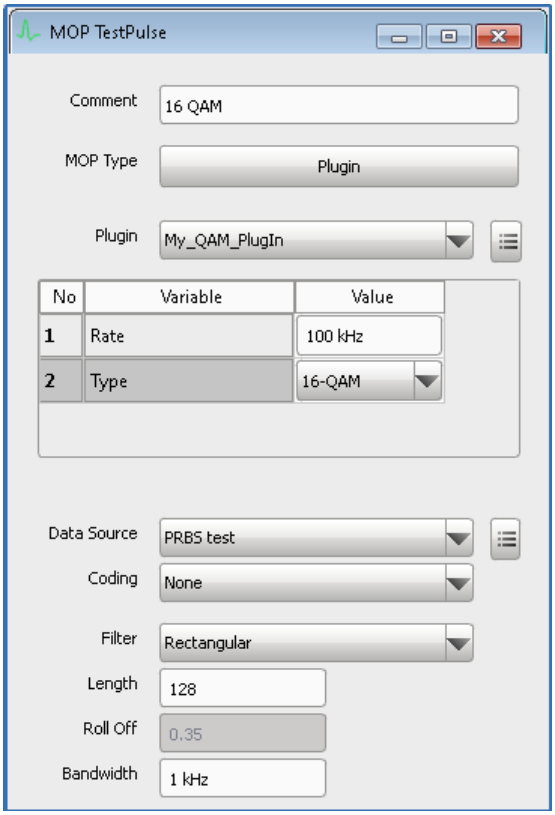
Remote command:

`PULSe:MOP:NOISe:BWIDth` on page 345

7.2.6.7 Plugins

Custom modulation schemes are used in the same way as the internal modulation schemes.

See chapter 16, "Defining Complex Modulation Schemes and IPM Profiles", on page 232.



Plugin , ... 94
Plugin Variables 94
Data Source, Coding, Filter 94

Plugin ,
Selects and loads a user-defined modulation from a file.

The modulation must exist in the "Plugin" library.

Use the standard context menu functions to:
- reset the plugin variables
- access the load "Plugin" settings, see chapter 16, "Defining Complex Modulation Schemes and IPM Profiles", on page 232

Remote command:
`PULSe:MOP:PLUGin:NAME` on page 278

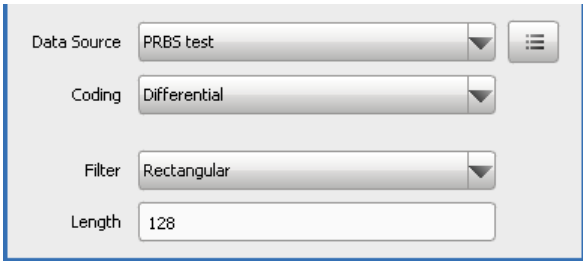
Plugin Variables
Resets all variables to the preset values defined in the plugin.

Remote command:
`PULSe:MOP:PLUGin:VARiable:CATalog?` on page 329
`PULSe:MOP:PLUGin:VARiable:SELect` on page 329
`PULSe:MOP:PLUGin:VARiable:VALue` on page 329

Data Source, Coding, Filter
See:
- chapter 15.1, "Data Sources Settings", on page 228

- "Coding" on page 95
- "Filter" on page 95

7.2.6.8 Data Source, Coding, Filter



The following setting are available for a subset of the provided MOP types:

Data Source 95
Coding 95
Filter 95

Data Source
Selects the data source for the custom modulation.

The data source has to be created first. See chapter 15, "Defining the Data Source", on page 226 for description of the provided settings.

Remote command:
`PULSe:MOP:DATA:DSRC` on page 342

Coding
Coding is a technique used to improve the signal properties and signal reception. In general, the coding schemes are applied prior to modulation, i.e. the modulation symbols are coded directly before I and Q values or frequency shifts are assigned.

This software supports the Differential and Grey coding and the combination of both.

For more information, see chapter A.4, "Common Coding Algorithms", on page 410.

Remote command:
`PULSe:MOP:DATA:CODing` on page 342

Filter
Filters are applied to shape the signal before it is modulated on the RF. The selected filter type and shape affect the output signal, especially while generating broadband signals.

This software supports the following filter types: "Rectangular", "Gauss", "Cosine" and "Root Cosine".

Depending on the selected filter type one or more filter parameters are provided for adjustment of the filter characteristic, like more steeper edges or customization of the transition bandwidth.

For more information, see chapter A.5, "Supported Filter and Impact of the Filter Parameters", on page 410.

Remote command:
PULSe:MOP:FILTer:TYPE on page 344
PULSe:MOP:FILTer:BT on page 343
PULSe:MOP:FILTer:BWIDth on page 343
PULSe:MOP:FILTer:LENGth on page 344
PULSe:MOP:FILTer:ROLLoff on page 344

7.2.7 Envelope and Modulation Graphs

Envelope Graph....96
Modulation Graph....97
└ Time domain display....97
└ I/Q constellation diagram....98
└ Spectrum....98

Envelope Graph
Displays the current pulse shape depending on the selected Standard Timing Profile and Rising/Falling Edge, Width, Rising/Falling Slope settings or Pulse Envelope Settings.

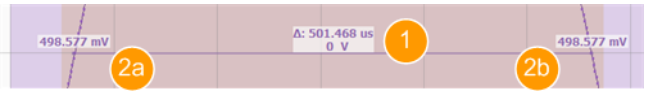
For example, see the envelope graph on figure 7-4.

"Normalized Pulse Envelope"
Changes the used units.
Available are "Watt", "dBm (W)" and "Voltage"

"Show Timing" Set "Show Timing > On" to visualize the pulse width, rise and fall time on the graph.
This visualization is useful, if you are not familiar with the definition of the pulse parameters when different profiles are used.
See also chapter 7.1, "Basics on Pulse Signals and Pulse Generation", on page 66.

"Temporary delta information"
Drag the mouse cursor on the graph.
The delta information is displayed temporary above the line.

Example: Measuring the pulse width with the help of delta marker



1 = Delta information: upper value is the Δx [s], lower value is the Δ y [units depending on the selected "Pulse Envelope Units"]
2a, 2b = Start and end y values [units depending on the selected "Pulse Envelope Units"]

Remote command:
PULSe:PREView:MODE on page 349

Modulation Graph

Displays the current modulation characteristics depending on the selected Modulation on Pulse (MOP) Settings.

The MOP is visualized on three diagrams: the time domain view, the spectrum and the I/Q constellation diagrams.

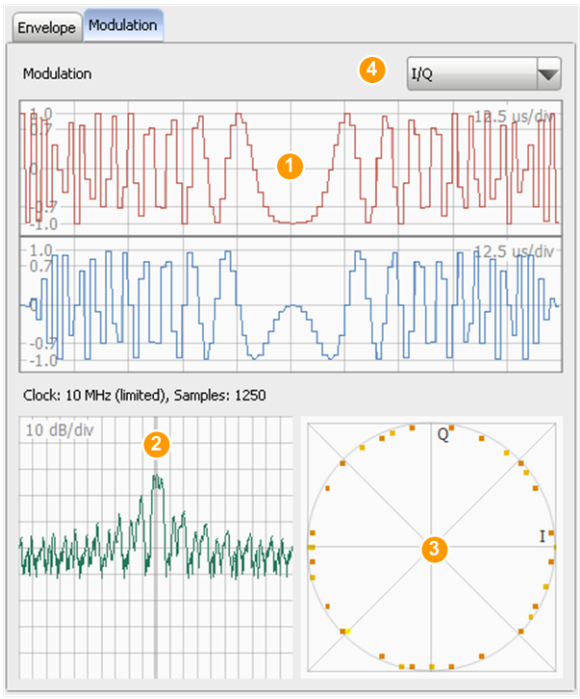


Fig. 7-12: Modulation Graph: Understanding the displayed information ("MOP Type > Polyphase code", "Type = P4", "Length M = 10")

1 = Time domain view
2 = Spectrum
3 = I/Q constellation diagram
4 = "Modulation > I/Q"

Time domain display ← Modulation Graph

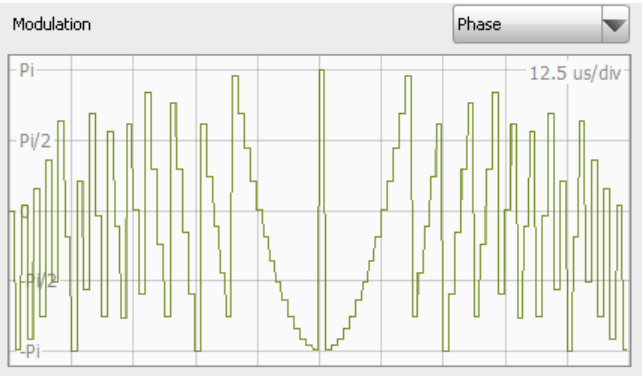
Use the "Modulation" parameter to change the displayed modulation characteristics.

Available are:

"I/Q diagram" Displays the I and Q signal versus time. Both signals use a linear scale in the range between -1.0 and +1.0.

"Phase diagram"

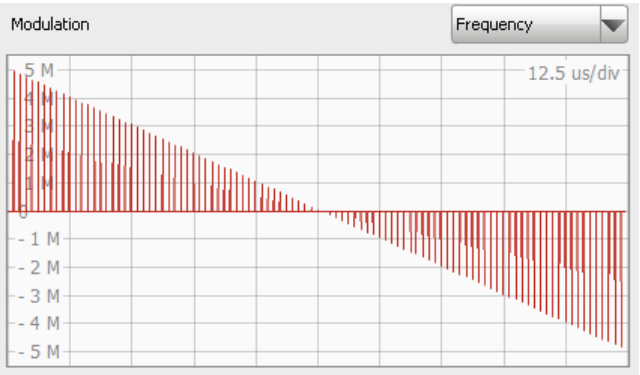
Displays the phase angle of the signal versus time. The diagram shows the phase changes in the range -π and +π.



A phase change is equivalent to a rotation at a constant radius in the constellation diagram.

"Frequency diagram"

Displays the instantaneous signal frequency versus time.



Remote command:
`PULSe:PREView:MOP` on page 350

I/Q constellation diagram ← Modulation Graph
Displays the probability at which I/Q points occur.

The color scale is logarithmic; the most probable points are indicated with bright yellow color.

Spectrum ← Modulation Graph
Displays the FFT spectrum. The FFT uses a logarithmic scale between 0 dB and -100 dB.

7.3 How to Create a New Pulse and Adjust Its Settings

Pulses are the fundamental building elements of any sequence.

See:

- "To create a new pulse" on page 99
- "To define the timing parameters of a pulse" on page 99
- "To define the level parameters of a pulse" on page 100
- "To define and apply modulation on the pulses (MOP)" on page 102
- "To generate a pulse with raised cosine envelope shape" on page 103
- "To define your custom envelope shape with an equation" on page 104
- "To visualize the pulse characteristics" on page 105

To create a new pulse

1. Perform one of the following:

   a) Select "Repository Tree > Pulse > New".
   b) In the "Sequence" dialog, press the pulse icon.

   A new pulse with default settings is created and added to the repository tree.
   New pulses are named `Pulse <n>` where n is a number starting at one.
   You can add information to describe and identify the pulse, like a name or a comment.

2. Change the pulse name, e.g. "P1".

   A pulse is described by its timing and level parameters, applied modulation on pulse (MOP) and enabled marker signals.
   For background information on the pulse parameters, see chapter 7.1, "Basics on Pulse Signals and Pulse Generation", on page 66.

   For description of the required settings, see:

   - chapter 7.2.2, "Pulse Timing Settings", on page 70
   - chapter 7.2.3, "Pulse Level Settings", on page 73
   - chapter 7.2.5, "Modulation on Pulse (MOP) Settings", on page 79
   - chapter 7.2.4, "Pulse Envelope Settings", on page 76
   - chapter 7.2.7, "Envelope and Modulation Graphs", on page 96

To define the timing parameters of a pulse

1. In the repository tree, select "Pulse > PulseName e.g. P1 > Timing".
2. Select "Standard Profile > On".
3. Select "Profile > Voltage 10/50/90".
4. Set the timing related parameters, such as delay, rise, fall, on, and off time, and the shape of the rising and falling slopes.
5. To configure the pulse repetition interval (PRI), use the parameter PRI.

6. Use the "Envelope" graph to visualize the current envelope profile.

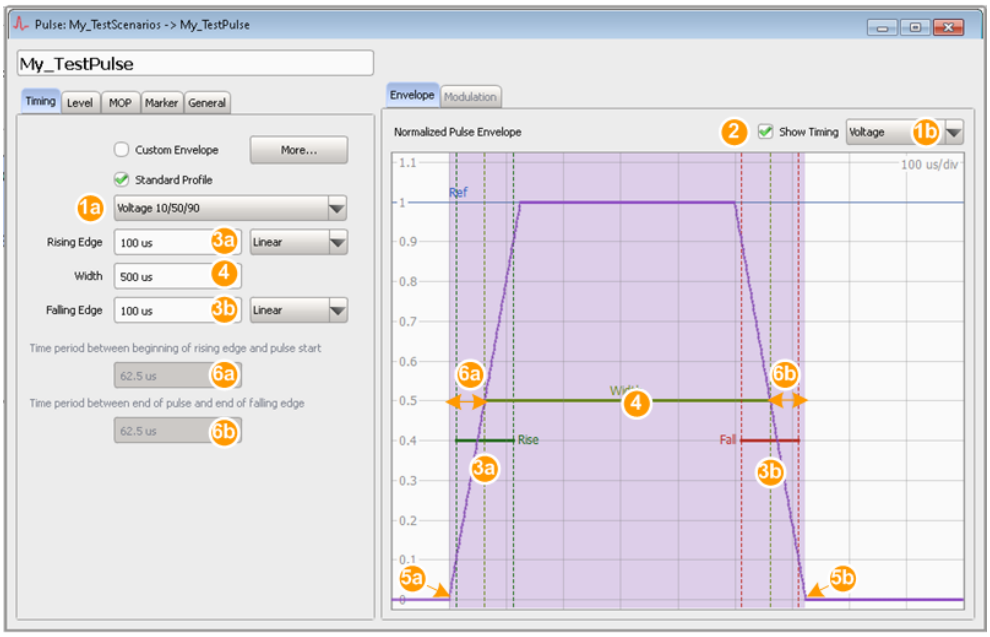


Fig. 7-13: Pulse Timing: understanding the displayed information

1a = "Standard Profile > Voltage 10/50/90"
1b = "Pulse Envelope Units > Voltage"
2 = "Show Timing > On" indicates the pulse timing parameters on the envelope graph
3a = "Rising Edge = 100 us", i.e. the time it takes the voltage to rise from 10% to 90% of the top level
3b = "Falling Edge = 100 us", i.e. the time it takes the voltage to fall from 90% to 10% of the top level
4 = "Width = 500 us", i.e. the duration the voltage is above 50 % of the top level
5a, 5b = Beginning of the rising and falling edge
6a = "Time period between beginning of the rising edge and pulse start = 62.5 us", i.e. the time it takes the voltage to rise from 0% to 50% of the top level
6b = "Time period between end of pulse and the end of falling edge = 62.5 us", i.e. the time it takes the voltage to fall from 50% to 0% of the top level

You can also import a custom shape or I/Q data, see:

- "To generate a pulse with raised cosine envelope shape" on page 103
- "To define your custom envelope shape with an equation" on page 104

Per default, the pulses within a pulse sequence, that is built by repeating the same pulse, have constant on time and PRI.

You can define IPM profiles to vary the pulse parameters on a pulse-to-pulse basis. See chapter 9.2, "How to Create IPM Profiles and Use Them to Vary Pulse Parameters", on page 144.

To define the level parameters of a pulse

Pulse level settings control the RF output power level during all pulse phases. The level is determined by two main settings, the "Top Power" and the "Base Power".

To access the level settings:

1. In the repository tree, select "Pulse > PulseName > Level".

   The tab comprises the pulse level settings, required to simulate a non-ideal pulse. The major reasons for non-ideal pulse shapes are the overshoot and ripple.

2. Change the "Level" settings and use the Envelope Graph to visualize the pulse characteristics.

   The "Envelope Graph" confirms that the envelope overshoots the "Top Power", experiences damped oscillations and amplitude "Droop" and settles to the final value.

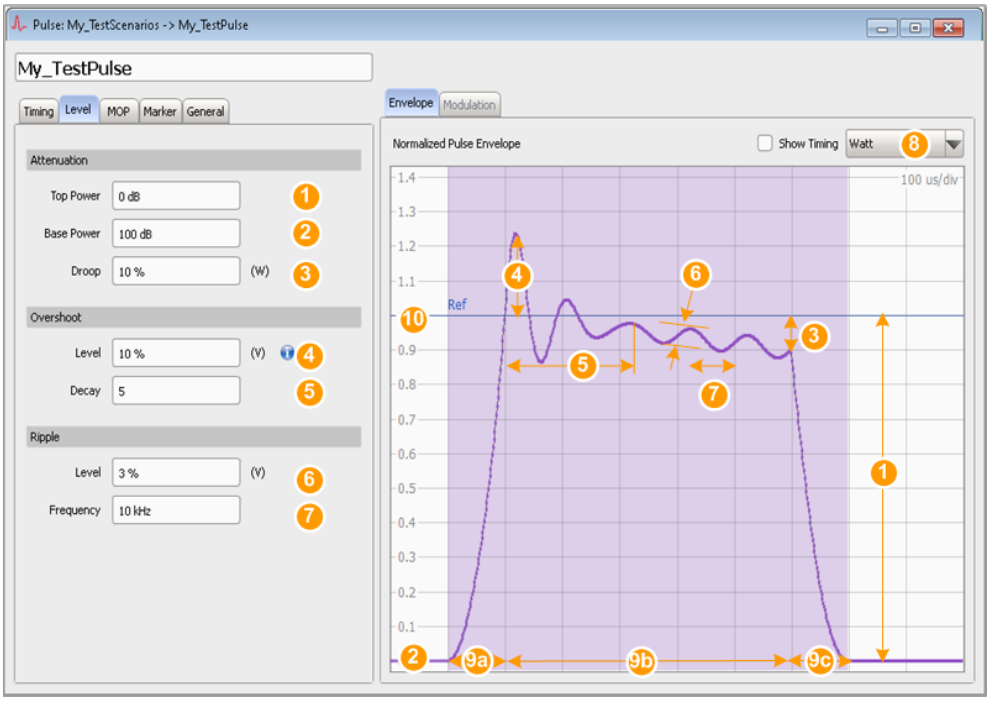


Fig. 7-14: Pulse Level: understanding the displayed information

1 = Pulse top power corresponds to 1000mW ("Pulse Envelope = Watt"), i.e. little or no attenuation ("Attenuation Top Power = 0 dB")
2 = "Base Power", i.e. high attenuation; typically 100 dB
3 = Amplitude change from the beginning to the end of the pulse ON time
4 = Overshoot level, in Volt (diagram uses Watts)
5 = Damping constant that influences the time it takes the overshoot to decay
6 = Amplitude of the oscillating signal, in Volt (diagram uses Watts)
7 = Signal oscillating with the selected frequency ("Frequency = 10 kHz", i.e. the oscillation period is 100 us)
8 = Pulse amplitude units (Y-axis units)
9a, 9b, 9c = Pulse timing: "Profile > Standard 0/100", "Rising Edge = Falling Edge = 100 us", "Width = 500 us" (see Standard Timing Profile)
10 = Reference value used by the calculation of the pulse level parameters: "Ref. = 0 dB (1 Watt)"; the value is set with the parameter Ref. Level

To define and apply modulation on the pulses (MOP)

The "MOP" settings define the intra pulse modulation, i.e. the modulation applied on the pulse rise, on, and off time. The R&S Pulse Sequencer provides a wide range of commonly used modulation schemes, such as AM, FM, or Chirp. You can also add your custom pulse content, chapter 16, "Defining Complex Modulation Schemes and IPM Profiles", on page 232.

Some "MOP" require data sources that are used with a modulation scheme. A data source for the symbols of the modulation has to be created first.

See chapter 15, "Defining the Data Source", on page 226 for description of the provided settings.

To access the "MOP" settings:

1. In the repository tree, select "Pulse > PulseName > MOP".
2. Select "MOP > Settings".

   The available modulation settings depend on the selected "MOP Type".
   See chapter 7.2.6, "Built-In Modulation Types and their Settings", on page 82.
3. Select for example "MOP > MOP Type > Chirp > Linear Chirp".

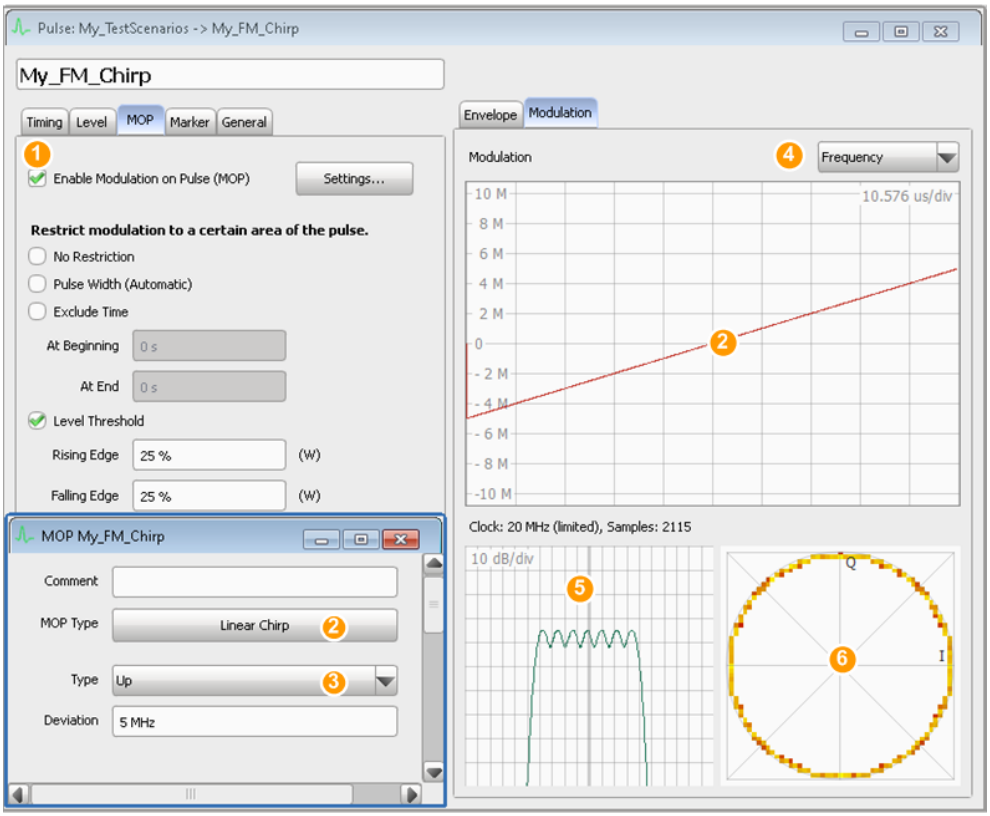


Fig. 7-15: Modulation on pulse (MOP): understanding the displayed information

1 = "Enable Modulation on Pulse (MOP) > On", i.e. MOP is applied
2 = Enabled is a liner chirp modulation; the diagram confirms the frequency variation
3 = "Type > Up", i.e. the frequency is increasing linearly

4 = Frequency diagram
5 = Spectrum
6 = Constellation diagram

For a description of the provided settings, see chapter 7.2.5, "Modulation on Pulse (MOP) Settings", on page 79.

4. Select "Enable Modulation on Pulse (MOP) > On".
5. Use the Restrict modulation to a certain area of the pulse settings to determine the exact pulse part the modulation is applied to, e.g.:

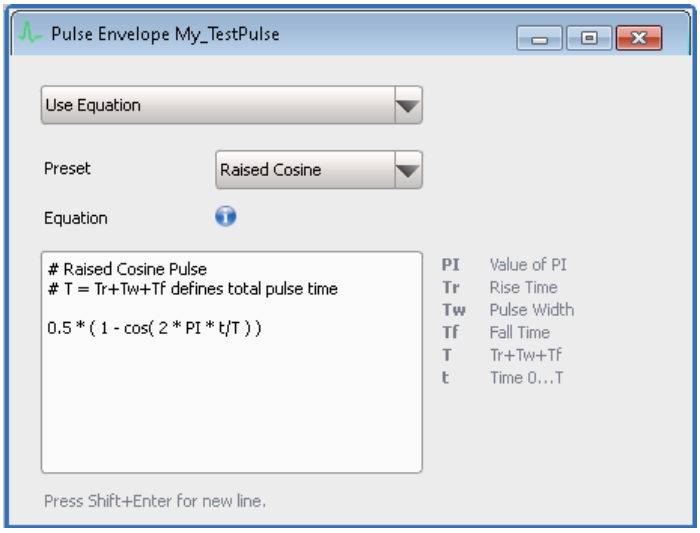
   a) Select "Level Threshold > On".
   b) Select "Rising Edge = Falling Edge = 25%".

   Use the "Waveform View" dialog to observe the effect of the enabled modulation, see chapter 18, "Visualizing and Analyzing Waveforms", on page 240.

To generate a pulse with raised cosine envelope shape

1. Select "Pulse > Timing"
2. Select "More" to customize the envelope.
3. Select "Use Equation".

   Envelope is described by an equation, as selected with the parameter "Preset".
4. Select "Preset > Raised Cosine"



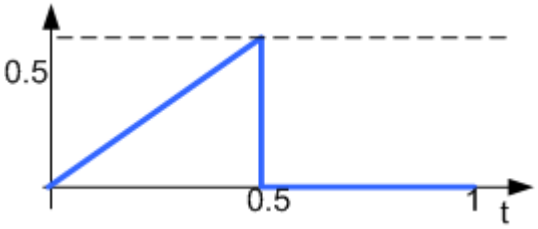
The dialog displays the used equation.
For overview of the available envelope shapes, see "Equations and parameters used to define custom pulse envelopes" on page 399.

5. Select "Pulse > Timing> Custom Envelope > On" to apply the envelope.

   The Envelope Graph displays the resulting envelope.

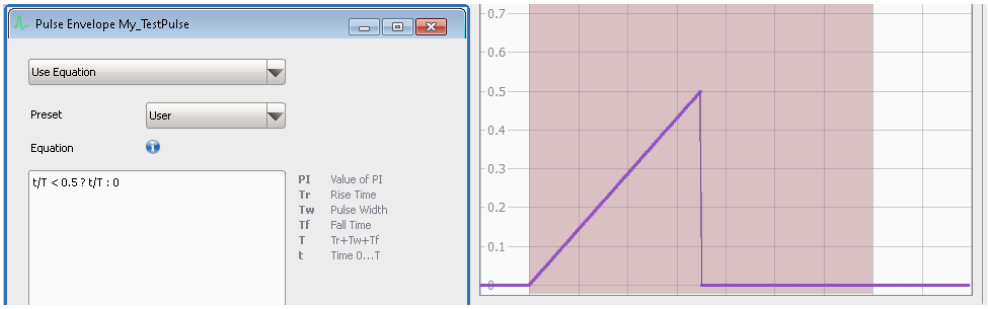
To define your custom envelope shape with an equation

This example explains how to enable an envelope shape that follows the shape on the following figure.



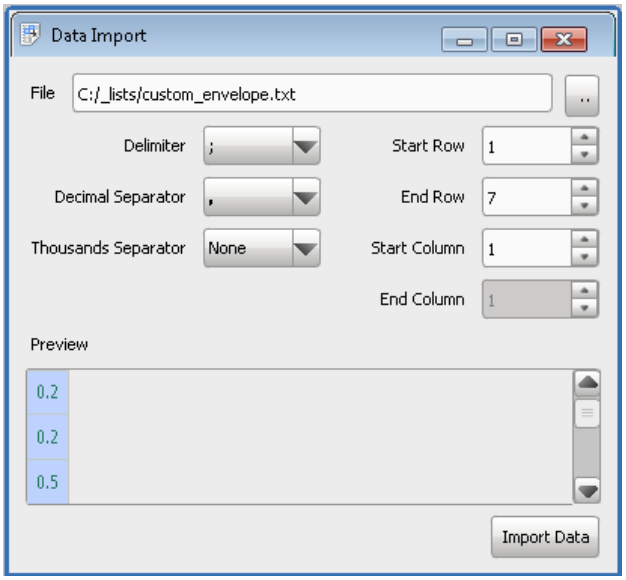
Equations are related to Voltage. Allowed are amplitude values between 0 and 1.

1. Select "Pulse > Timing > More" and select "Use Equation".
2. Select "Preset > User" and enter the equation:
   `t/T < 0.5 ? t/T : 0`
   See also chapter A.2, "Formula Syntax", on page 398.
3. Select "Envelope > Normalized Pulse Envelope > Voltage" and compare the resulting shape.



To generate an envelope shape in a table form

1. Select "Pulse > Timing > More" and select "Use Imported Data".
2. Select "Import Data from File", navigate to the file describing the envelope and load it.



To visualize the pulse characteristics

1. Use the "Modulation Graph" to visualize the configured modulation schemes.
   See "Modulation Graph" on page 97.
2. Use the "Envelope Graph" to display the envelope shape.
   See figure 7-13.
3. Set "Show Timing > On" to visualize the pulse width, rise and fall time on the graph.

8 Building Pulse Sequences

This section explains the sequencing principles and signal processing, like creating loops and defining overlaying signals.

For information on sequences containing waveforms, see chapter 12, "Working with Waveforms and Generating Interfering Signals", on page 192.

8.1 About the Sequencing Principles

Sequences are described in table form where the rows are processed sequentially and appended to the final waveform, see figure 8-1.



Fig. 8-1: Default segments processing: One segment after the other

Each row is repeated once or as defined with the repetition count parameter (see "Rep. Cont." on page 110).

See also:

- "To define and enable pulse repetition" on page 118
- chapter 8.2.2, "Pulse Repetition Settings", on page 112

Complex sequences may also contain loops, overlays and fillers:

- **Loop**
  A loop spans one or more rows (pulses) and processes these rows in the same order several times. Nested loops are complex loops that span rows and loops. The loop duration is defined by the loop repetition number. The loop repetition number is a fixed value or a randomly selected value within a value range, defined with its minimum, maximum and step values.
  See also:
  – "To define and apply a loop" on page 119
  – chapter 8.2.3, "Loop Settings", on page 113
- **Overlay**
  An overlay processes rows in parallel. The row processing start at the same time and lasts up to the overlay duration is. Longer segments are truncated.
  Overlay is a method that changes the default behavior (figure 8-1) and adds several pulses (segments) to a common time line; i.e. segments are stacked on top of each other. An overlay group spans several segments of any kind, like regular pulses and fillers.
  Segments are added to the existing data starting at the beginning of the first segment. If the new segment is shorter than the overlay length, the remainder is left blank. Longer segments are truncated, see figure 8-2.

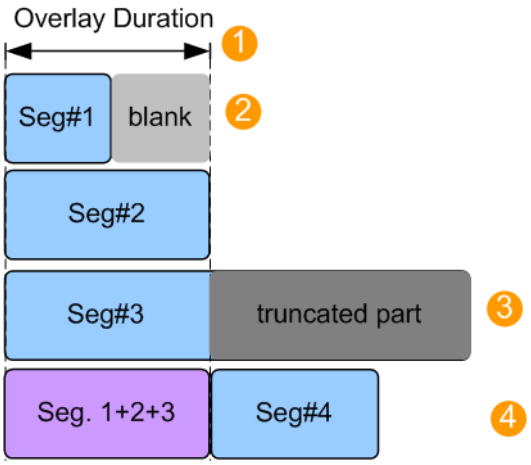


Fig. 8-2: Segments processing in overlay mode: Segments are added

1 = Overlay Duration: sets the length of the resulting sum segment
2 = Short segment: remainder is left blank
3 = Long segment: segment is truncated
4 = Resulting sum segment (Seg#1+Seg#2+Seg#3): Seg#4 is processed right after the overlay duration elapses

See also:
- "To define and enable overlaying segments" on page 122
- "To include a sub-sequence in an existing overlay" on page 124
- chapter 8.2.4, "Overlay Settings", on page 114

- **Sub-sequence**
  A sub-sequence defines a sequential sequence of segments within an overlay.
  See also:
  - "To include a sub-sequence in an existing overlay" on page 124
- **Filler**
  Fillers are unmodulated continuous wave (CW) or blank signals used to fill in certain period in a sequence.
  There are two ways to set the duration of a filler: by defining the time duration to be filled in (total duration) or by defining of a certain moment of time (time synchronization).
  The filler duration can be a fixed value or as an equation, where the equation uses the loop variables, if the filler is configured within a loop.
  See also:
  - "To create and add a CW segment" on page 122
  - chapter 8.2.5, "Fillers Settings", on page 114

8.2 Sequence Settings

To access these settings:

► In the repository tree, select "Sequence > New".

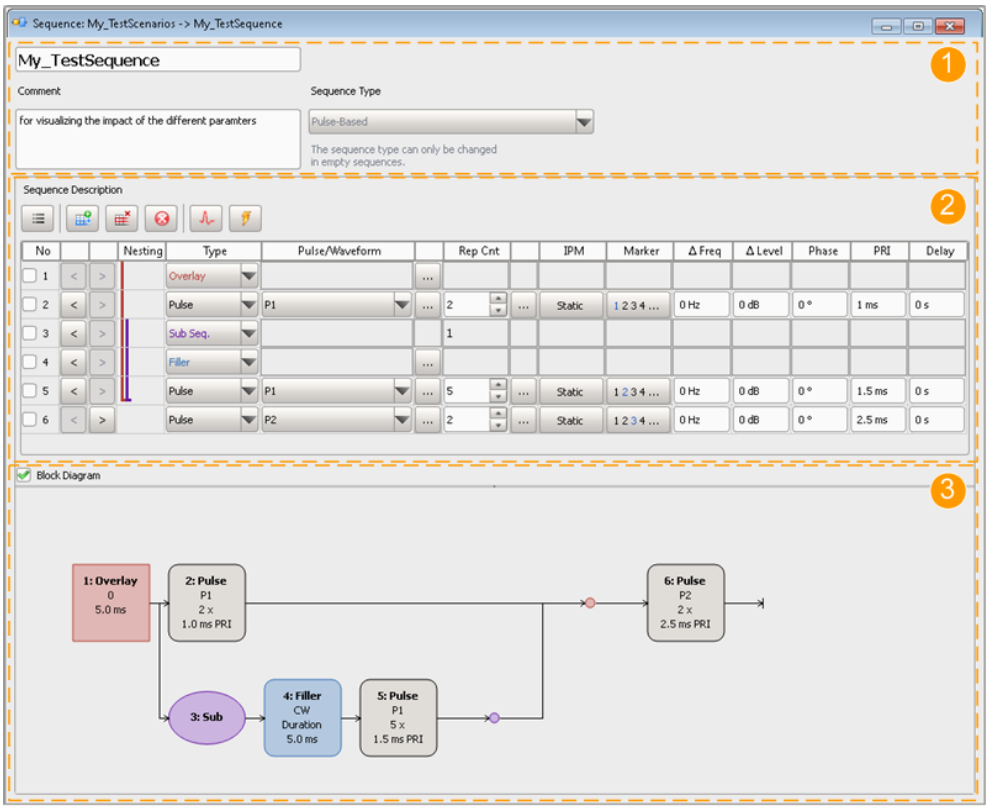


Fig. 8-3: Sequence Settings

1 = Sequence name, comment and type
2 = Sequence description table, where each row represents on item
3 = Interactive graphical representation if the current sequence, displayed if "Block Diagram > On"

For step-by-step descriptions, see chapter 8.3, "How to Create Sequences and Use the Control Elements", on page 117.

The following settings are available:

- Sequence Description Settings 109
- Pulse Repetition Settings 112
- Loop Settings 113
- Overlay Settings 114
- Fillers Settings 114
- Lists with Multiple Sequences 116

8.2.1 Sequence Description Settings

Sequence Name, Comment and Type, Sequence Description Table, Block Diagram 109
New, Insert, Append, Remove, Clear Items 109
Select, No. 110
Nesting 110
Type 110
Pulse/Waveform 110
Rep. Cont. 110
... (more) 111
IPM 111
Marker 111
Δ Freq. 111
Δ Level 111
Phase 111
PRI 111
Delay 111

Sequence Name, Comment and Type, Sequence Description Table, Block Diagram

A sequence is defined with its unique name, an optional description and with information on the sequence content, waveforms or pulses.

A sequence is defined in table form where each row represents one item.

If enabled, an interactive block diagram displays the current sequence configuration.

- Left click to select the item (see "Select, No." on page 110).
- Right click to access the context menu (see "New, Insert, Append, Remove, Clear Items" on page 109).
- A double click on an item works like the "More (...)" icon in the "Sequence Description Table" and accesses the settings of the item directly.

Remote command:

`SEQuence:CREate` on page 277
`SEQuence:CATalog?` on page 276
`SEQuence:SELect` on page 277
`SEQuence:NAME` on page 278
`SEQuence:COMMent` on page 278
`SEQuence:TYPE` on page 375
`SEQuence:REMove` on page 279

New, Insert, Append, Remove, Clear Items

Standard functions for items handling, see "Standard function in the context menus" on page 24.

You have to select a row for example to delete it or to insert a row before it.

Remote command:

`SEQuence:ITEM:ADD` on page 279
`SEQuence:ITEM:CLEar` on page 282
`SEQuence:ITEM:DELete` on page 281

Select, No.
Select an item to perform any row based actions.

Items are indicated by consecutive number; also used for indication in remote control.

Remote command:
`SEQuence:ITEM:SELect` on page 280
`SEQuence:ITEM:COUNt?` on page 280

Nesting
Indents table rows in order to include them in a loop, to assign them to an overlay or build parallel branches.

See "To define and enable overlaying segments" on page 122.

Remote command:
`SEQuence:ITEM:INDent` on page 376

Type
Defines the type of content of the current item. Available are:

"Pulse/Wave"	To select a pulse or a waveform, use the Pulse/Waveform.
"Filler"	Creates dummy data, like a blank segment, or a CW signal. See chapter 8.2.5, "Fillers Settings", on page 114.
"Loop"	Creates a loop See chapter 8.2.3, "Loop Settings", on page 113.
"Overlay"	An overlay processes sequence items in parallel See chapter 8.2.4, "Overlay Settings", on page 114.
"Sub. Seq"	In overlay, includes a sequence in a sequence, i.e. creates a parallel branch containing sequently processed items. See To include a sub-sequence in an existing overlay.

Remote command:
`SEQuence:ITEM:TYPE` on page 381

Pulse/Waveform
Lists all available pulses and wavefroms.

- To access the settings of an existing pulse, select "... (more)".
- To crate a new pulse, select the pulse icon.
  See also chapter 7, "Creating a Pulse Library", on page 66.

Remote command:
`SEQuence:ITEM:PULSe` on page 379
`SEQuence:ITEM:WAVeform` on page 379

Rep. Cont.
Sets the number of times the item is repeated. For more complex repetition patterns, select "... (more)".

See chapter 8.2.2, "Pulse Repetition Settings", on page 112.

Remote command:
`SEQuence:ITEM:REP:COUNt:FIXed` on page 380
`SEQuence:ITEM:LOOP:COUNt:FIXed` on page 377

... (more)
Accesses a dialog with further settings.

IPM
Enables and defines an inter-pulse modulation.

See chapter 9, "Defining and Enabling Inter-Pulse Modulation Effects", on page 127.

Marker
Defines the active markers per pulse phase, see chapter 17, "Defining and Enabling Marker Signals", on page 236.

Δ Freq.
Enables a frequency offset relative to the reference frequency.

Frequency dependent parameter values are defined relative to this value.

See also "To define and apply a loop" on page 119.

Remote command:
`SEQuence:ITEM:FREQuency:OFFSet` on page 376

Δ Level
Defines a level offset per row.

A negative level offset simulates a pulse attenuation. For example, a "Level = -3 dB" means that the pulses of the current row are processed with power level that is the half of the power.

Level dependent parameter values are defined relative to this value.

See also "To define and apply a loop" on page 119.

Remote command:
`SEQuence:ITEM:LEVel:OFFSet` on page 376

Phase
Sets a phase offset at the beginning of the row. Phase dependent parameter values are defined relative to this value.

For example, a "Phase = 30 deg" shifts the phase in the very first pulse of the current row; enabled modulation would further change the phase. The phase offset is not compensated at the row end.

Remote command:
`SEQuence:ITEM:PHASe:OFFSet` on page 378

PRI
Sets the Pulse Repetition Interval (PRI), i.e. the pulse period (see figure 7-1).

Use the "PRI" parameter to define pulses with constant total pulse cycle duration and enabled timing variations like jitter for example.

Remote command:
`SEQuence:ITEM:PRI` on page 379

Delay
Enables a start delay.

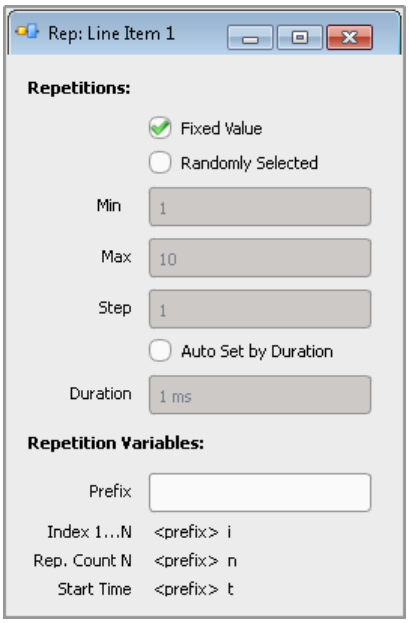
A delay within an overlay shifts the start time of the overlay elements relative to each other. Time dependent parameter values are defined relative to this value.

See also "To define and apply a loop" on page 119.

Remote command:
`SEQuence:ITEM:PDELay` on page 378

8.2.2 Pulse Repetition Settings

Per default, a table row in the sequence description table, i.e. a pulse or a waveform, is processed once. To repeat an item several times, enable a repetition.



For a step-by-step instruction, see "To define and enable pulse repetition" on page 118.

Repetition Number
Defines how many times a pulse, a waveform or a loop will be repeated.

The repetition number can be:
- a fixed value
- a randomly selected value within a value range, defined with its minimum, maximum and step values ("Randomly Selected")
- defined to fit in a selected duration ("Auto Set by Duration"), where the repetition number can be rounded up or down

Remote command:
`SEQuence:ITEM:REP:TYPE` on page 381
`SEQuence:ITEM:REP:COUNt:MINimum` on page 380
`SEQuence:ITEM:REP:COUNt:MAXimum` on page 380
`SEQuence:ITEM:REP:COUNt:STEP` on page 380
`SEQuence:ITEM:REP:COUNt:DURation` on page 379
`SEQuence:ITEM:REP:COUNt:ROUNding` on page 380

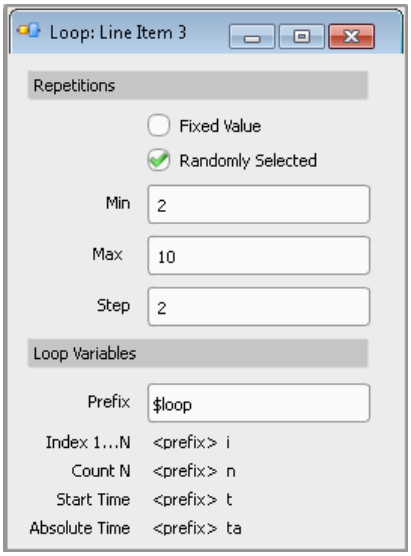
Repetition Variables
You can define repetition variables that will be used by the pulse calculation within one row.

"Prefix" Indicates the repetition the variable belongs to. If empty, no variables are used.

Remote command:
`SEQuence:ITEM:REP:VARiable` on page 381

8.2.3 Loop Settings

Per default, table rows in the sequence description table are processed sequentially. To process a subset of pulses or waveforms several times, create a loop.



For a step-by-step instruction, see "To define and apply a loop" on page 119.

Loop Repetition
The loop repetition number is a fixed value or a randomly selected value within a value range, defined with its minimum, maximum and step values.

Remote command:
`SEQuence:ITEM:LOOP:TYPE` on page 377
`SEQuence:ITEM:LOOP:COUNt:MINimum` on page 377
`SEQuence:ITEM:LOOP:COUNt:MAXimum` on page 377
`SEQuence:ITEM:LOOP:COUNt:STEP` on page 377

Loop Variables
You can define loop variables that will be used by the pulse calculation within the loop.

"Prefix" Indicates the loop to that the variable belongs. If empty, no loop variables are used.

"Loop Index 1 .. N"
"<prefix>_i" indicates the current repetition number and is a value in the range 1 .. N, where N is the repetition number.

"Loop Count N" "<prefix>_n" indicates the repetition number, fixed or randomly selected as defined with Loop Repetition.

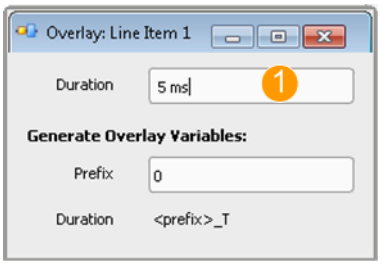
"Start Time" "<prefix>_t" indicates the start time of each subsequent loop

"Absolute Time" "<prefix>_ta" indicates the absolute time

Remote command:
`SEQuence:ITEM:LOOP:VARiable` on page 378

8.2.4 Overlay Settings

An overlay processes rows in parallel. The row processing starts at the same time and lasts up to the overlay duration is. Longer items are truncated.



1 = Overlay Duration: sets the length of the resulting sum item

For a step-by-step instruction, see "To define and enable overlaying segments" on page 122.

Duration
Sets the duration of the sum item.

Remote command:
`SEQuence:ITEM:OVL:WTIMe` on page 378

Overlay Variables
You can define overlay variables that will be used by the pulse calculation.

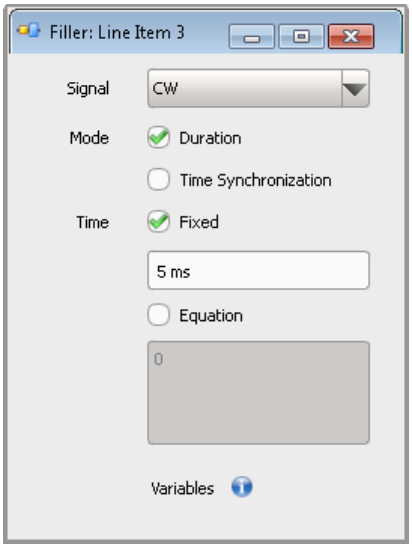
"Prefix" Indicates the overlay to that the variable belongs. If empty, no overlay variables are used.

"Duration" "<prefix>_t" indicates the overlay duration.

Remote command:
`SEQuence:ITEM:OVL:VARiable` on page 378

8.2.5 Fillers Settings

You can enable and add fillers.



For a step-by-step instruction, see "To create and add a CW segment" on page 122.

Signal
Selects the type of signal to be generated for filling in the dummy segment.

"Blank" A blank segment with the selected duration is appended to the sequence.

"CW" A segment containing a continuous wave (CW), i.e. unmodulated signal, is used.

"Hold" Holds the last available pulse sample.

Remote command:
`SEQuence:ITEM:FILLer:SIGNal` on page 375

Mode
Filling segments are commonly used for synchronization purposes. This parameter defines the way the filler duration is determined:

"Duration" Enables a dummy signal for a given period of time.

"Time Synchronization"
In this mode, signal data is generated up to the defined point in time. If the selected time has already elapsed an error is generated and the calculation is stopped.
Use this mode if a signal shall start at a defined point in time and the length of the previous signals vary or is not known.

Remote command:
`SEQuence:ITEM:FILLer:MODE` on page 375

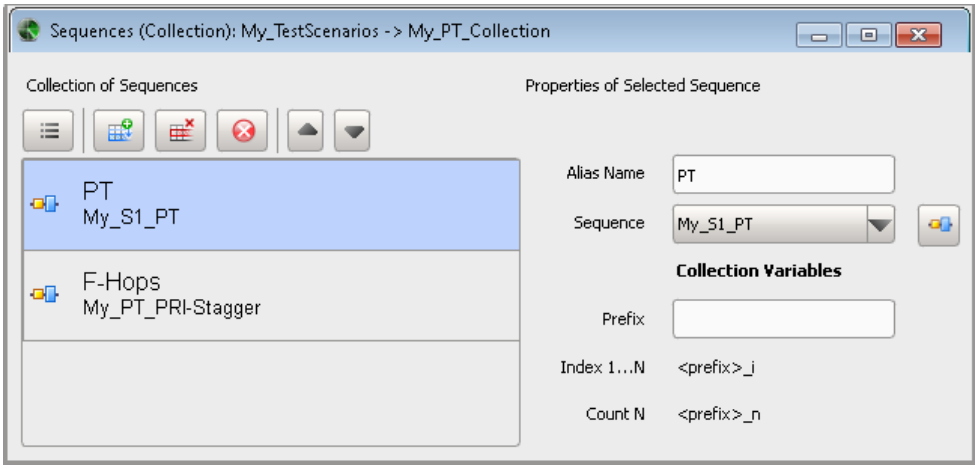
Time
Defines the timing of the filler.

"Fixed" Uses a fixed time

"Equation" Defines the point of time as an equation.

Remote command:
SEQuence:ITEM:FILLer:TIME on page 375
SEQuence:ITEM:FILLer:TIME:FIXed on page 376
SEQuence:ITEM:FILLer:TIME:EQUation on page 375

8.2.6 Lists with Multiple Sequences



Switching between multiple sequences is possible in the pulse train sequential scenario.

List of sequences
Displays a list of sequences.

To add a sequence, select the "Data Source" icon and select "Insert After".

To reorder sequences (items in the list), use the "Up/Down" functions.

Use the standard "Append", "Remove Last", or "Delete All" functions to add or remove a sequence. The provided settings are self-explanatory.

Remote command:
SCENario:CSEQuence:ADD on page 279
SCENario:CSEQuence:SELect on page 280
SCENario:CSEQuence:CURRent on page 363
SCENario:CSEQuence:CLEar on page 282
SCENario:CSEQuence:DELete on page 281

Name
Enters a name.

Remote command:
SCENario:CSEQuence:ALIas on page 363

Sequence
Select a sequence from the list of available sequences.

Remote command:
SCENario:CSEQuence on page 368

Collection Variables
You can define collection variables and use them e.g. in the formula editors, in the reports, etc.

"Prefix" Indicates the sequence to that the variable belongs. If empty, no variables are used.

"Index 1 .. N" "<prefix>_i" indicates the current number and is a value in the range 1 .. N.

"Count N" "<prefix>_n" is the number of variables.

Remote command:
SCENario:CSEQuence:VARiable on page 368

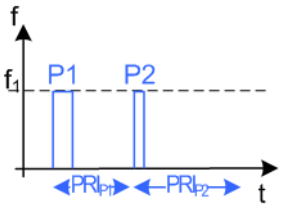
8.3 How to Create Sequences and Use the Control Elements

See:

- "To create a simple sequence" on page 117
- "To create and add a CW segment" on page 122
- "To define and enable pulse repetition" on page 118
- "To define and enable overlaying segments" on page 122
- "To include a sub-sequence in an existing overlay" on page 124
- "To define and apply a loop" on page 119
- "To create a scenario with several pulse train sequences" on page 125

To create a simple sequence

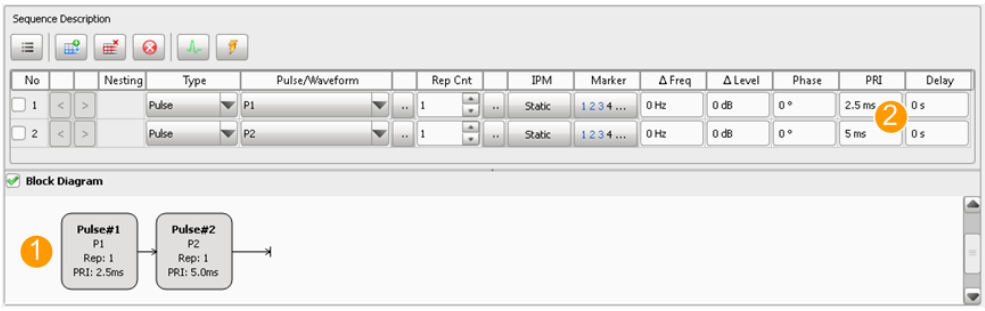
This examples explains how to create a simple sequence with two pulses, P1 and P2.



We assume that the required pulses have been created, see chapter 7.3, "How to Create a New Pulse and Adjust Its Settings", on page 98.

1. In the repository tree, select "Sequence > New".

2. In the "Sequence" dialog, enter a "Name" and a "Comment", e.g. `My_S1_PT`.
3. For each sequence item, perform the following steps:
   a) Select the "Append New Item" icon.
   b) In the "Sequence Description" table, select "Type > Pulse" and select an existing pulse, e.g. "Pulse > P1".
   c) If the list is empty, select the pulse icon and create a pulse first.
   d) See chapter 7.3, "How to Create a New Pulse and Adjust Its Settings", on page 98
   e) Set the PRI_{P1}, e.g. "PRI = 2.5 ms".
4. Enable "Show Block Diagram".



1 = Sequence with two items
2 = Pulse with different PRIs, i.e items with different duration

The items are processed once and one after the other (see also figure 8-1). You can change the default processing, see:

- "To define and enable pulse repetition" on page 118
- "To define and enable overlaying segments" on page 122
- "To define and apply a loop" on page 119

To define and enable pulse repetition

This examples explains how to repeat pulses several time, for example to configure a pulse sequence as illustrated on figure 8-4.

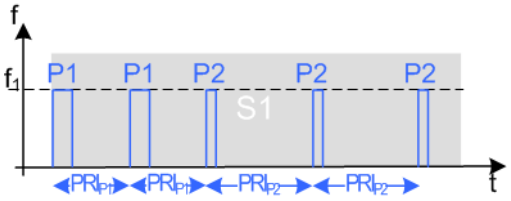


Fig. 8-4: Example of pulse sequence composed of 2xP1 and 3xP2, where the pulses use different PRIs

PRI_{P1} = 2.5 ms
PRI_{P2} = 5 ms

We assume that a sequence with two items has been created, see "To create a simple sequence" on page 117.

1. Open this sequence.

2. In the "Sequence Description" table, for the first pulse, select "Rep. Count = 2"
3. For the second item, set "Rep. Count = 3".



The pulse sequence S1 consists of 5 pulses and has a PRI_{S1} = 20ms.



Use the"Waveform View" display to visualize the resulting waveform, see chapter 18, "Visualizing and Analyzing Waveforms", on page 240.

To define and apply a loop

This examples explains how to create a loop with pulses and to repeat it several times, for example to configure a pulse sequence as illustrated on figure 8-5.

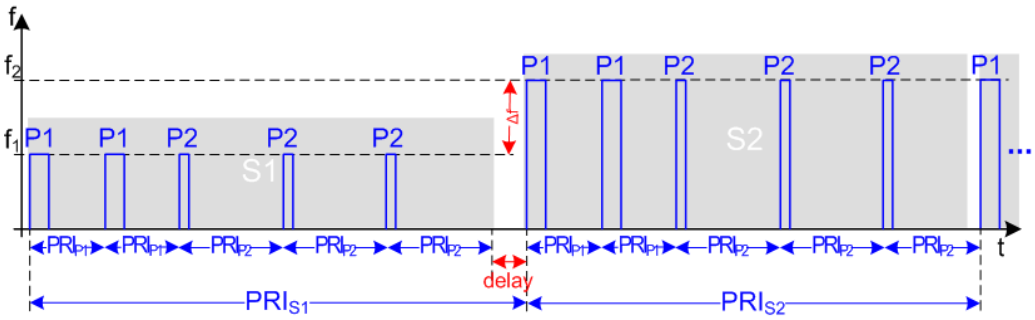
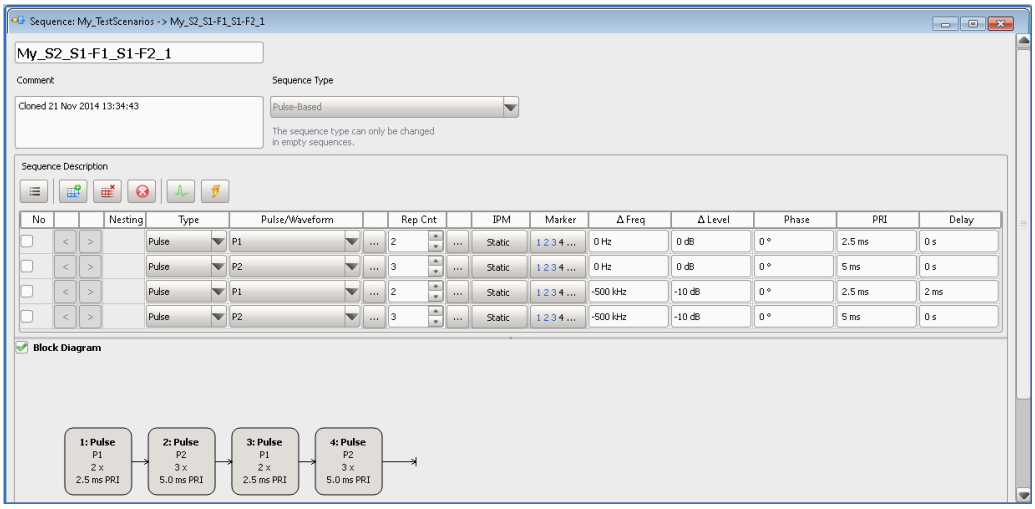


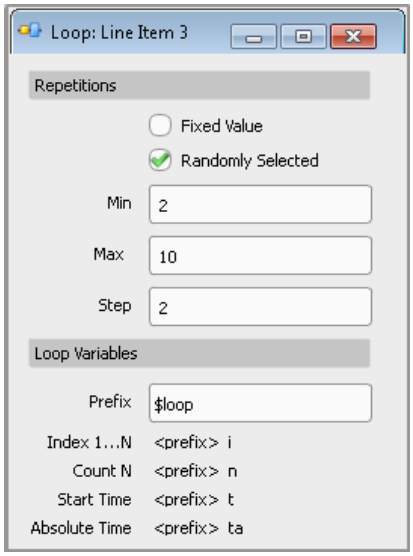
Fig. 8-5: Example of pulse sequence composed of two sub-sequences, S1 and S2

S1, S2 = Both sequence follow the same pulse pattern (2xP1 and 3xP2)
PRI_{S1} = PRI_{S2} + "Delay"
f_2 = f_1 + "ΔFreq"

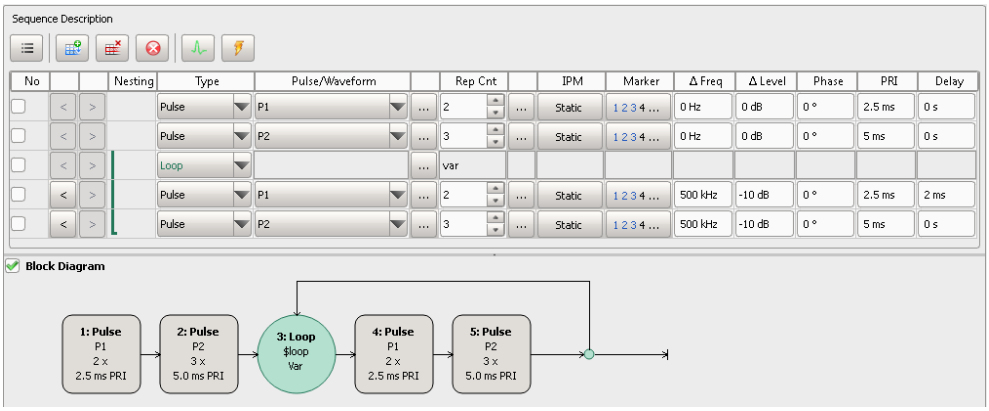
We assume that a sequence with four items has been created, as shown on the following figure.



1. In the "Sequence Description" table:
   a) select the third item and select "Insert Item Before Selection"
   b) Select "Type > Loop"
   c) select the "... (more)" icon and enable a random loop repetition.



   The loop will be repeated 2, 4, 6, 8 or 10 times.
   d) to include the last two items (Line items 3 and 5) in the loop, select the "Indent" icon.



The sequence graph in the "Sequence Description" confirms that the two items are included in the loop. The loop is repeated random number of times.

2. To enable a different PRI for the looped items, select the first pulse in the loop and set "Delay = 2 ms", i.e. $PRI_{S1} = PRI_{S2} + 2$ ms = 22 ms.
3. For the pulses in the loop, set the "Δ Freq = 500 KHz" and "Δ Level = -10 dB".
4. Assign the sequence `My_S2_S1-F1_S1-F2` to a pulse train scenario, start the waveform calculation and select "Volatile > View".

   The "Data View" confirms that the waveform comprises of two sub-sequences with different PRI, where the second one is repeated four times.

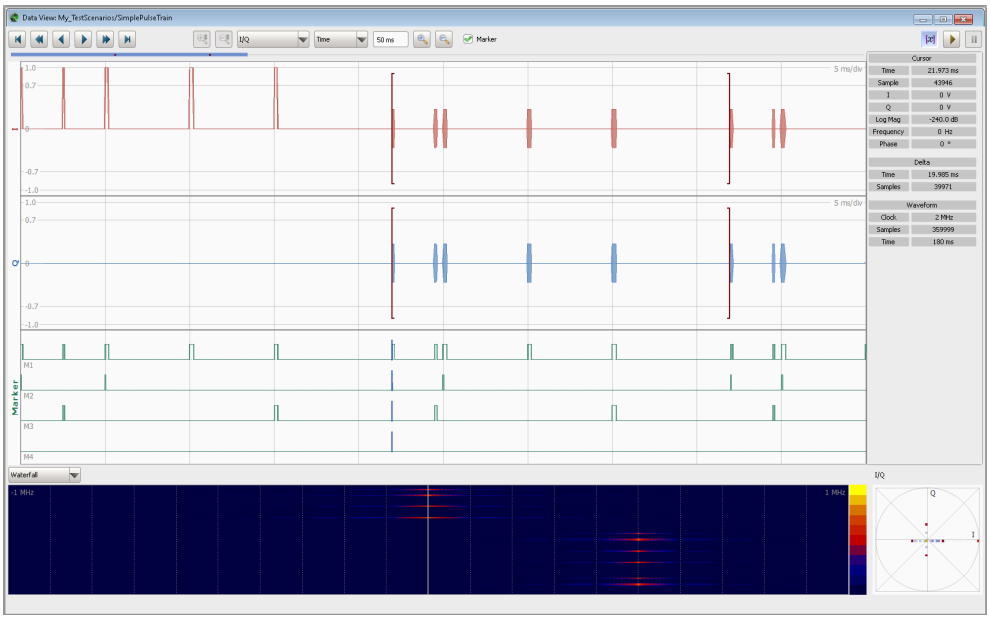
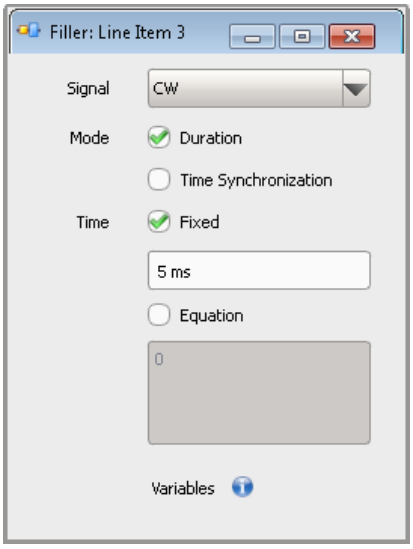


Fig. 8-6: Data View: Example of a sequence

Select "View Mode > Frequency" and "View Mode > Average Power" to observe the frequency and level values.
For more information, see figure 18-1.

To create and add a CW segment

1. Open an existing sequence.
2. In the "Sequence Description" table, add a new item, select "Type > Filler" and select "... (more)".
3. In the "Configure Filler" dialog:
   a) select "Signal > CW"
   b) select "Mode > Duration"
   c) select "Time > Fixed = 5 ms"



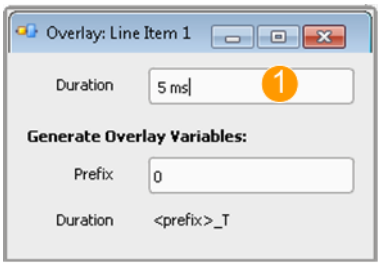
To define and enable overlaying segments

This examples explains how to overlay three items (see figure 8-2).

We assume that the a sequence with four items has been created.

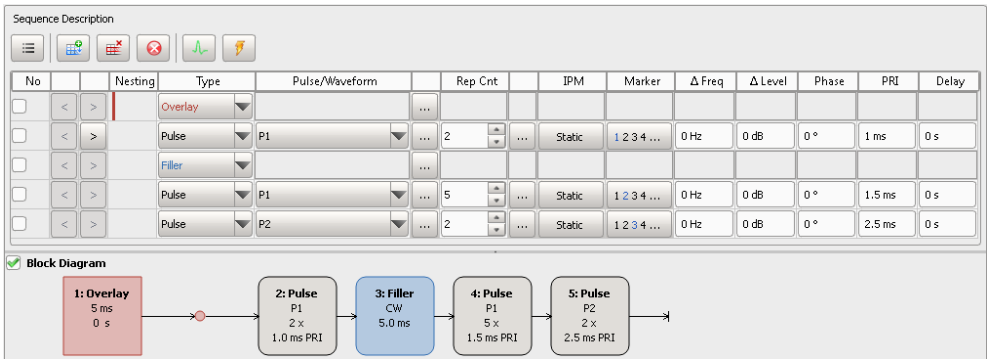
No			Nesting	Type	Pulse/Waveform		Rep Cnt		IPM	Marker	Δ Freq	Δ Level	Phase	PRI	Delay
1	<	>		Pulse	P1	..	2	..	Static	1 2 3 4 ...	0 Hz	0 dB	0 °	1 ms	0 s
2	<	>		Filler		..									
3	<	>		Pulse	P1	..	5	..	Static	1 2 3 4 ...	0 Hz	0 dB	0 °	1.5 ms	0 s
4	<	>		Pulse	P2	..	2	..	Static	1 2 3 4 ...	0 Hz	0 dB	0 °	2.5 ms	0 s

1. In the "Sequence Description" table of this sequence:
   a) select the first item
   b) select the "New/Select" icon and select "Insert Item Before Selection"
   c) for the appended (first item), select "Type > Overlay" and select "... (more)".
2. In the "Configure Overlay" dialog, select "Overlay > Duration = 5 ms".

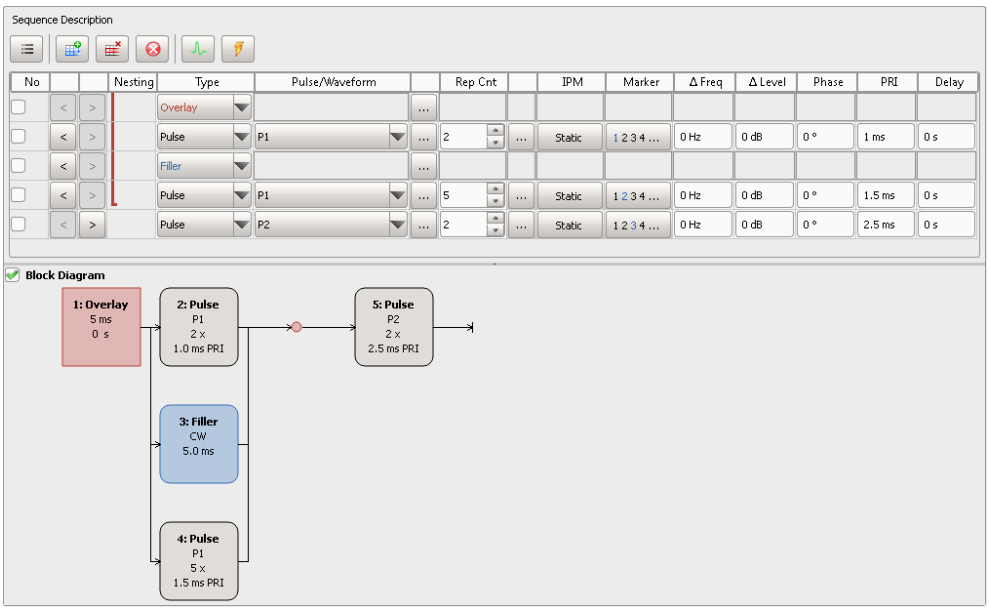


1 = Overlay Duration: sets the length of the resulting sum item

The block diagram confirms that an overlay is created. There are still no segments added to it.



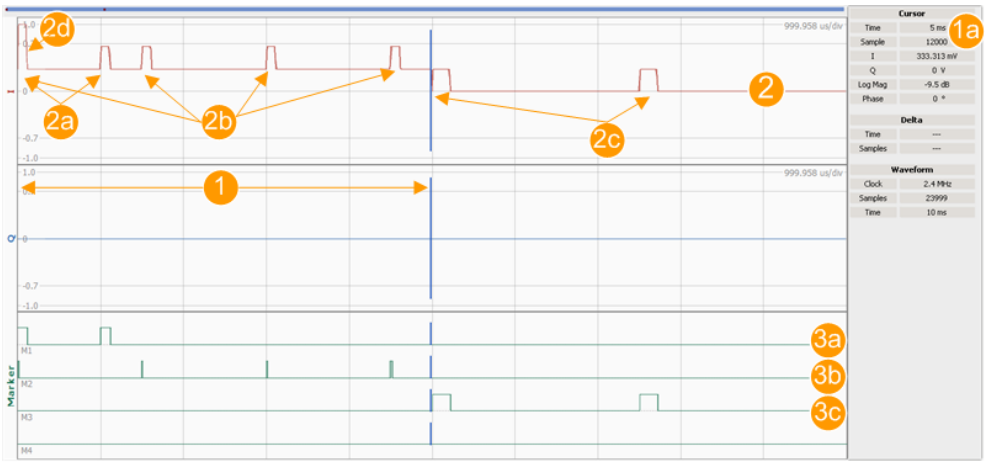
3. To include the first three items (Line items 2, 3 and 4) in the overlay, select the "Indent" icon.



The block diagram confirms that the three items are overlaid; the fourth item will be processed after elapsing of the overlay duration.

4. To observe the effect of an enabled overlay, open the "Data View" dialog.

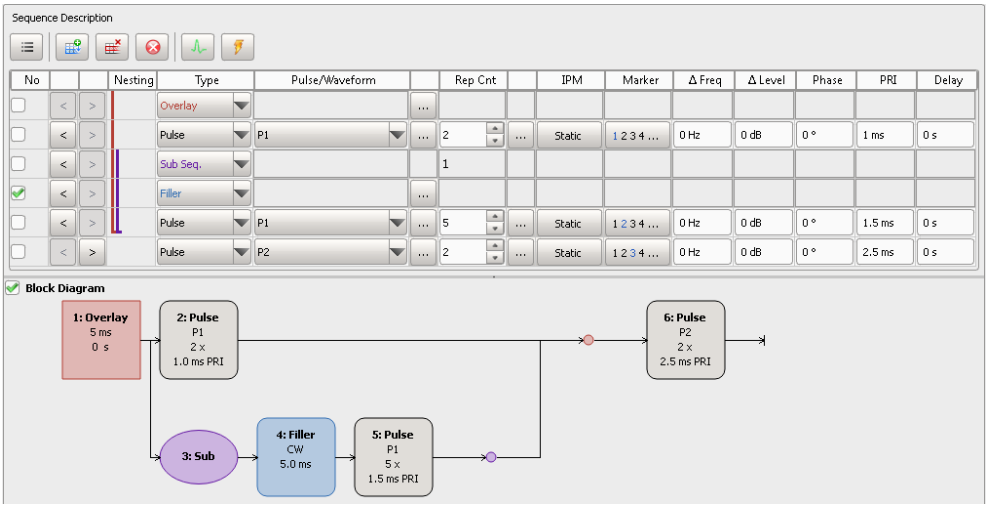
See also chapter 18.3, "How to Analyze the Content of Waveform Files and Files with I/Q Data", on page 245.



1 = Overlay
1a = Overlay "Duration = 5 ms"
2 = Generated pulse sequence = sum segment followed by the item outside of the overlay
2a+2b = Resulting sum segment
2a = Short segment: 2x pulse P1 with "PRI = 1 ms" (remainder is left blank)
2b = Long segment: 4x pulse P1 with "PRI = 1.5 ms" (segment is truncated)
2c = Outside segment: 2xP2 (appended right after the overlay)
2d = The first P1 (short segment) and the first P2 (long segment) are superimposed
2a, 3b, 3c = Marker traces ("M1 = M2 = M3 = All"): indicate the pulse On time of each of the pulse

To include a sub-sequence in an existing overlay

1. Open the sequence you have created with "To define and enable overlaying segments" on page 122.
2. In the "Sequence Description" table, insert a new item after the second one and select "Type > Sub Sequence".
3. To build a parallel branch with the items 4 and 5, use the "Indent" icon.



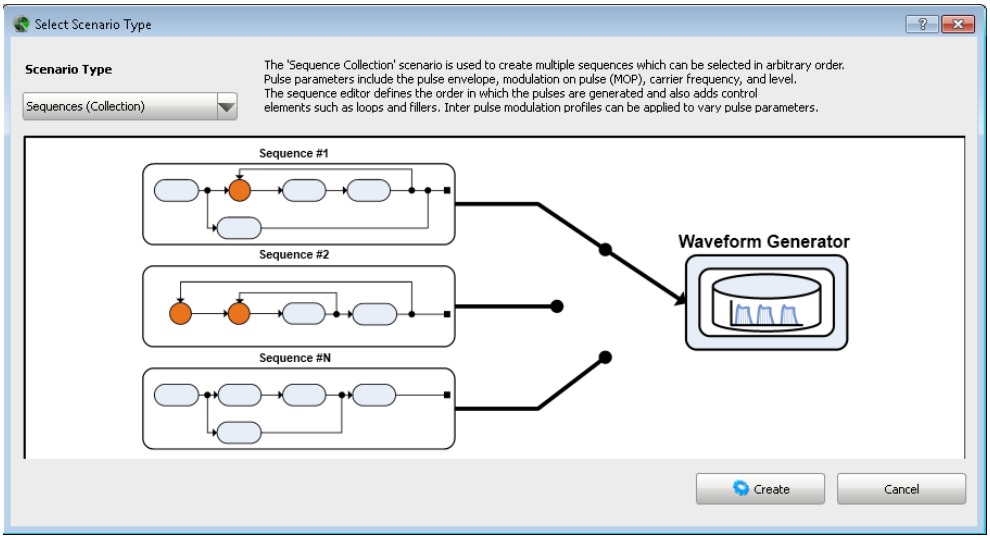
The block diagram confirms that the items are overlaid, where the second parallel branch consists of two subsequently processed pulses; the fourth item will be processed after elapsing of the overlay duration.

To create a scenario with several pulse train sequences

If you test situation requires to test the receiver's ability to correctly detect different pulse sequences, you can use "Sequences (Collection)" scenario. In this dedicated scenario, you can configure several pulse train sequences and switch between them sequentially.

The following example creates a complex scenario with two pulse train sequences to explain the principle.

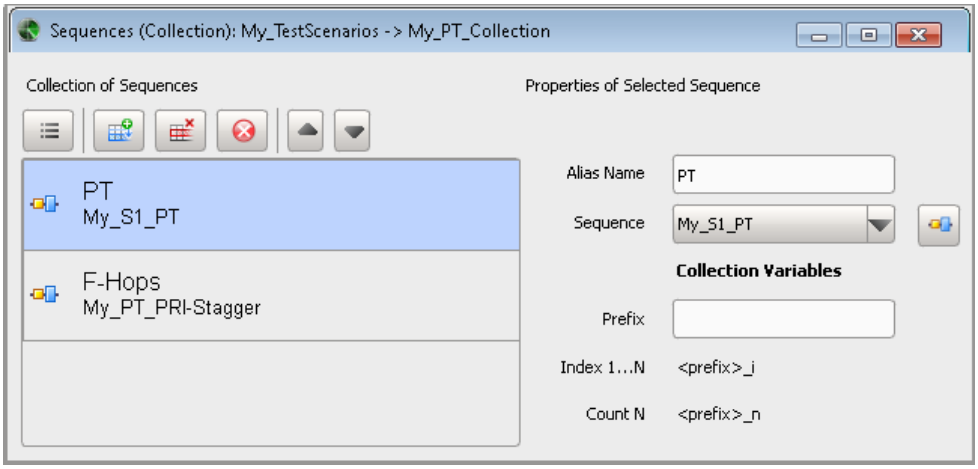
1. Select "Repository Tree > Scenario > New".
2. In the "Scenario Wizard", select "Scenario Type > Sequences (Collection)".


Select Scenario Type
Scenario Type
Sequences (Collection)
The 'Sequence Collection' scenario is used to create multiple sequences which can be selected in arbitrary order.
Pulse parameters include the pulse envelope, modulation on pulse (MOP), carrier frequency, and level.
The sequence editor defines the order in which the pulses are generated and also adds control
elements such as loops and fillers. Inter pulse modulation profiles can be applied to vary pulse parameters.
Sequence #1
Sequence #2
Sequence #N
Waveform Generator
Create
Cancel

3. Confirm with "Create".
4. Enter a scenario name and add a description.
5. In the block diagram, select "Sequences > Edit".

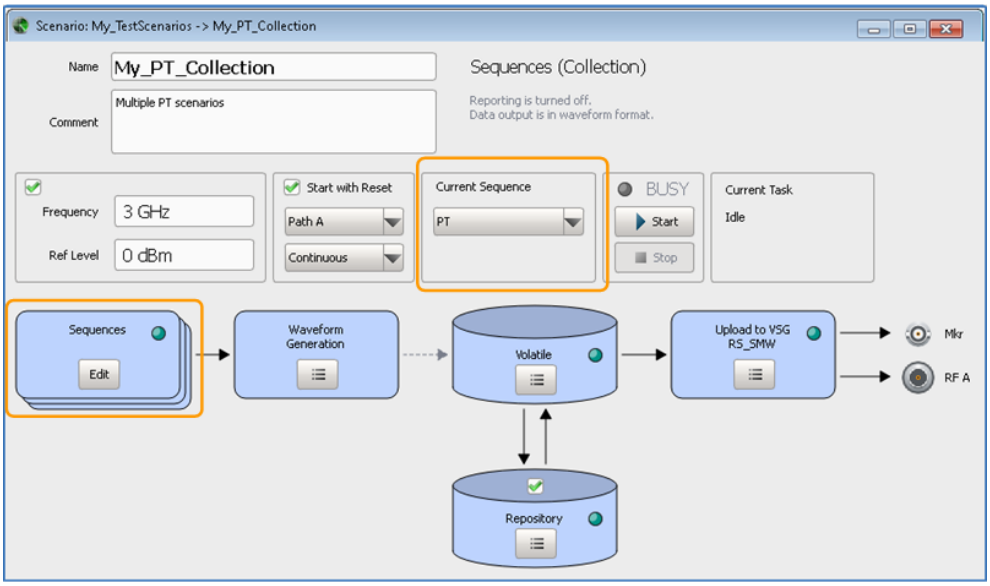
   The "Multiple Sequences" dialog opens.
6. Use the standard "New" function and insert the first sequence in the list.

   You can select from the list of available sequences or create a new one.
   See also "To create a simple sequence" on page 117.
7. In the list of sequences, select a "Sequence" and use the standard "New", "Append/Remove" and "Up/Down" functions to add new items to the list and reorder them.



For description of the provided settings, see chapter 8.2.6, "Lists with Multiple Sequences", on page 116.

8. In the "Scenario" dialog, select the name of the sequence that is currently used, e.g. "PT".



For description of the provided scenario settings, see chapter 6.2, "Scenario Settings", on page 54.

9 Defining and Enabling Inter-Pulse Modulation Effects

Per default, pulses calculated from the settings of one table row in the sequence description table are identical. To simulate a change of parameters from one pulse to one another, like PRI stagger for example, you can define and enable inter-pulse modulation (IPM) effects and assign them to the sequence elements.

Throughout this description, the IPM effects are commonly named parameter variations.

Pulse repetition patterns

The pulse repetition patterns are commonly used to:

- Avoid range ambiguities caused by echoes of far away objects, which fall into the detection period of the next pulse
- Protect the radar signal against jamming
- Deal with clutter (ground, sea, weather)
- Make the radar more difficult to be identified by frequency hopping

IPM effects

In the following, two common IPM technics, the Jittered PRI and Staggered PRI, are used to explain the principle:

- *Jittered PRI*
  In general, a jitter is understood as the short-term variations of a significant pulse parameter, deviating from their ideal positions in time in either random or ordered way (see figure 9-1).
  You can, for example, use the random variation of the rising and falling pulse edge positions to simulate a technically imperfect trigger signal.

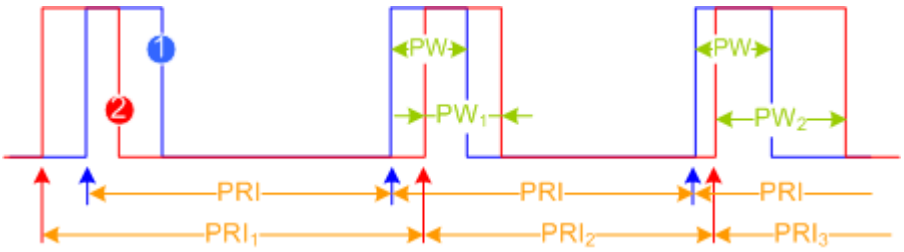


Fig. 9-1: Example of Jitter effects

1 = Ideal pulse with constant pulse repetition interval (PRI) and pulse width (PW)
2 = Pulse affected by Jitters: Jitter PRI ($PRI_1 \neq PRI_2$) and varying pulse width ($PW_1 \neq PW_2$)

  As illustrated on figure 9-1, the time between two subsequent pulses varies randomly, on a pulse-to-pulse basis, as does the pulse width.

- *Staggered PRI*
  Staggered pulse trains are commonly used against range ambiguities und blind speeds, see also "Pulse repetition patterns" on page 127.

Staggered PRI is a transmission method based on pulse trains with constant pulse width (PW) and varying PRI. The principle of staggered PRI is illustrated on figure 9-2

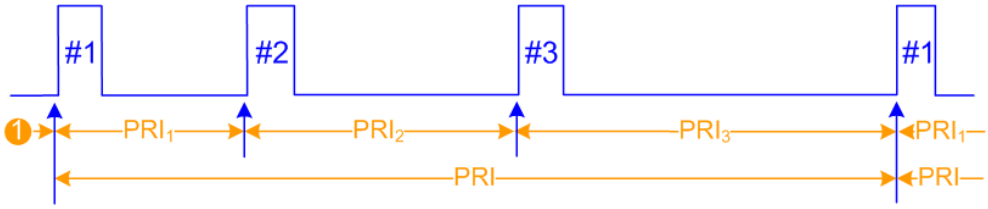


Fig. 9-2: Example of an IPM effect: Staggered PRI

1 = Stagger start delay
#1,#2,#3 = Pulse train group composed of three pulses
PRI = Group PRI: Pulse repetition interval of the pulse train group
PRI_1,PRI_2,PRI_3 = Pulses transmitted with different PRI

As illustrated on figure 9-2, within a pulse train group each successive pulse uses a different PRI; the pulse train group is slewing.

- *Frequency Hopping*
  Frequency hopping and frequency agility is a further anti-jamming method. If frequency hopping is used, the radar signal switches rapidly and periodically between different carrier frequencies.

There are no predefined parameter variations in the software but IPM profiles can be assigned to the parameter listed in table 9-1.

Table 9-1: Pulse parameters to that IPM profiles can be assigned

Parameter group	Pulse Parameter
Level	Overshoot [%]
	Offset [dB] Δ Level
	Attenuation Top [dB] Attenuation > Top Power
	Attenuation Base [dB] Attenuation > Base Power
	Droop [%]
	Ripple [%] Ripple > Level
	Ripple Frequency [Hz]
Modulation	AM Modulation Depth [%]
	AM Frequency [Hz]
	FM Deviation [Hz]
	FM Frequency [Hz]
	Chirp Deviation [Hz]

Parameter group	Pulse Parameter
Timing	Rise Time [s] Pulse > Time > Rising Edge
	Pulse Width [s] Pulse > Time > Width
	Fall Time [s] Pulse > Time > Falling Edge
	Delay [s]
	PRI [s]
	PRF [Hz] Pulse repetition frequency, calculated from the selected PRI
Phase	Offset [°]
Δ Freq.	Offset [Hz]

Source profiles

Parameter variations are described as the combination of a source profile and the target element, pulse or sequence, the variation is assigned to. If you assign more than one variations to an element, these variations are superimposed (see chapter 9.1.4, "Configure Inter-Pulse Modulation (IPM) Settings", on page 142).

Each jitter and IPM effect affects one of the parameters listed in table 9-1 and can use the following profiles:

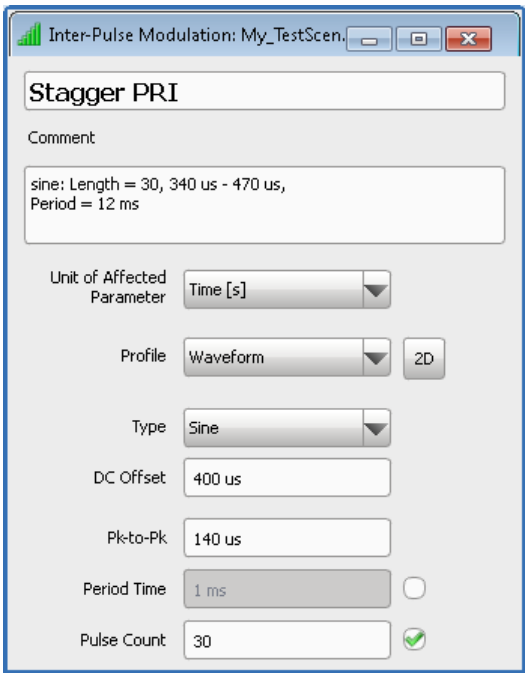
- Steps
- Waveform
- Random List
- List
- Interpolated Shape
- Random
- Equation

Source profiles are jitter and IPM profiles. You can use the same profiles for more than one variation; the R&S Pulse Sequencer calculates the profile values independent from each other.

9.1 IPM Profiles Settings

To access these settings:

► Select "Repository Tree > IPM > New".



Pulse variations are defined by:

- Common IPM Settings 130
- IPM Profiles Settings 132
- Edit List Settings 140
- Configure Inter-Pulse Modulation (IPM) Settings 142

9.1.1 Common IPM Settings

IPM Profile Name 131
IPM Profile Comment 131
Unit of Affected Parameter 131
Profile 131
2D 131

IPM Profile Name
Enters the name of the profile.

Remote command:
IPM:CREate on page 277
IPM:NAME on page 278
IPM:CATalog? on page 276
IPM:SELect on page 277
IPM:REMove on page 278

IPM Profile Comment
Adds a description.

Remote command:
IPM:COMMent on page 278

Unit of Affected Parameter
Sets the units of the IPM parameters. Available are: "Time [s]", "Frequency [Hz]","Level [dB]", "Phase [deg]".

Remote command:
IPM:UNIT on page 313

Profile
Sets the shape of the profile:
- Steps
- Waveform
- Random List
- List
- Interpolated Shape
- Random
- Equation

Further available parameters depend on the selected shape.

Remote command:
IPM:TYPE on page 313

2D
Opens a preview diagram of the selected profile.

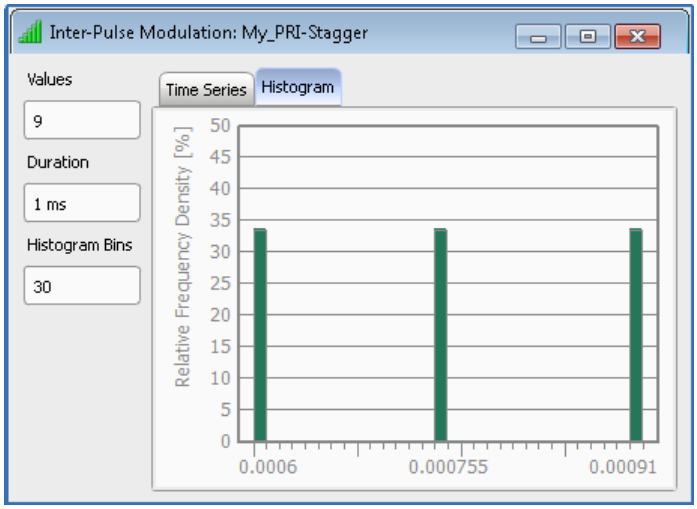
Two views are available:
- "Time Series"
  that is a visualization of the profile variation over time
- "Histogram"
  that is a statistical representation of the relative frequency density, calculated for the selected number of "Histogram Bins".
  The histogram plots the number of times data points appear in a particular bin.

The displayed characteristics on both views are defined by:

"Values" Sets the number of values to be displayed.

"Duration" Sets the displayed duration.

"Histogram Bins"



Sets the number of histogram bins, i.e. the histogram sets used to group the obtained data.
Used are bins with equal size, where the bin size is calculated form the value range used on the y axis in the time series display and the number of bins.

9.1.2 IPM Profiles Settings

The following IPM shapes can be defined:

Steps....132
List....133
Waveform....134
Interpolated Shape....135
Equation....136
Random List....136
Random Steps....137
Random....138
Plugin....139

Steps
This IPM profile follows a staircase shape and creates a sequence of identical values before it moves on the next one.

The "Burst Length" defines often a certain step is repeated, i.e. how many identical values are created. The profile shape is defined as function of "Number of Steps", "Start" and "Increment" values; the stop value is calculated automatically.

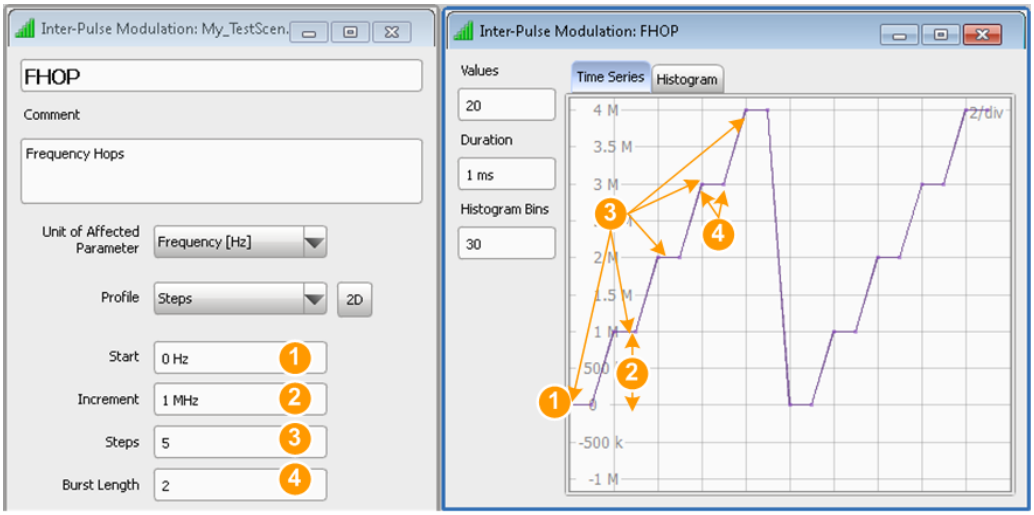


Fig. 9-3: IPM with shape "Profile = Steps"

1 = "Start" = 0 Hz
2 = "Increment" = 1 MHz
3 = "Number of Steps" = 5
4 = "Burst Length" = 2

Remote command:
`IPM:STEP:BURSt` on page 318
`IPM:STEP:STEPs` on page 319
`IPM:STEP:STARt` on page 319
`IPM:STEP:INCRement` on page 318

List

The IPM shape is a sequence of discrete values defined in a list. The list items are processed sequentially and cyclically; when the list end is reached, the list is processed from the beginning.

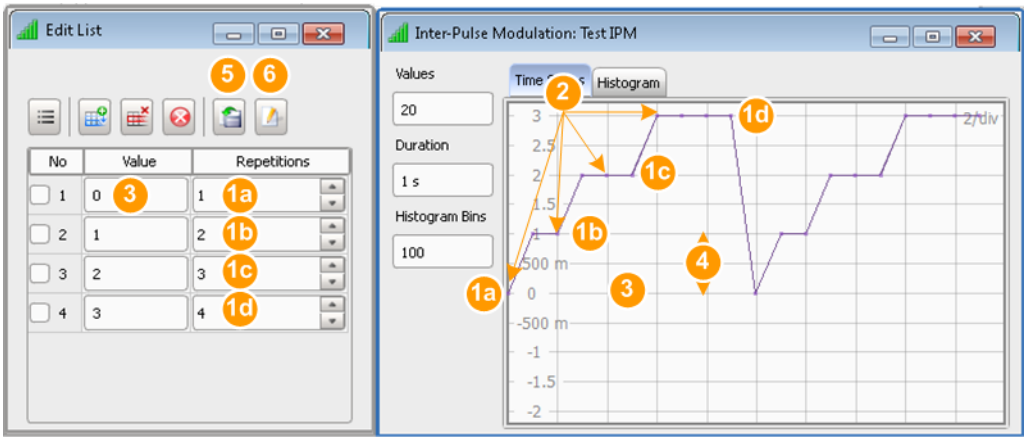


Fig. 9-4: IPM with shape "Profile = List"

1a, 1b, 1c, 1d, 1e = The increments are repeated different number of times: "Count#1 = 1", "Count#2 = 2", "Count#3 = 1", etc.
2 = "List Count" = 4
3 = "List Start" = 0

4 = "List Increment" = 1
5 = "Import Data from File" icon
6 = "Populate list" icon

"List" Accesses the standard list editor to define a list in table form, see "Edit List" on page 140.

- To create a list manually, use the standard icons and functions in the context menu (see "Standard function in the context menus" on page 24).
  Enter the list values manually.
- To import a list in ASCII format, select the "Import Data from File" icon.
- To create a list automatically and fill it in with values, select the "Populate list" icon.
  Enter the "Start" and "Increment" values and the number of list elements ("Count") and select "Populate".

Remote command:
`IPM:LIST:ITEM:ADD` on page 279
`IPM:LIST:ITEM:COUNt?` on page 280
`IPM:LIST:ITEM:SELect` on page 280
`IPM:LIST:ITEM:VALue` on page 314
`IPM:LIST:ITEM:REPetition` on page 314
`IPM:LIST:ITEM:DELete` on page 281

Waveform
Sets an IPM shape that follows a ramp, sine or triangular waveform function. The waveform amplitude is defined by the "Peak-to-Peak" value, a constant "Offset" can be applied to the entire waveform; the waveform period is set as "Period Time" or as number of pulses ("Pulse Count") it lasts.

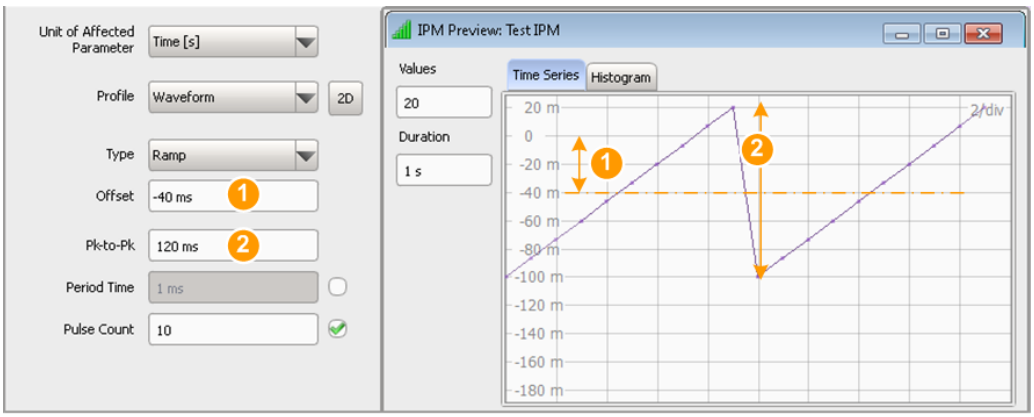


Fig. 9-5: IPM with shape "Profile = Waveform" (Type = Ramp)

1 = "Offset" = -40 ms
2 = "Peak-to-Peak" = 120 ms

"Linear Ramp" The linear ramp is a profile with a sawtooth shape.
The generated values monotonically increase from ("Offset"-"Pk-to-Pk"/2) to ("Offset"+"Pk-to-Pk"/2).

"Sine" The sine profile creates values that follow one period of a sine wave. Enable a "Phase Offset" to change the start phase of the sine wave.

"Triangular" The triangular profile changes a parameter from a minimum to a maximum value and back to the minimum value, following a triangular shape.

Remote command:
`IPM:WAVeform:BASE` on page 320
`IPM:WAVeform:TYPE` on page 320
`IPM:WAVeform:PKPK` on page 321
`IPM:WAVeform:OFFSet` on page 320
`IPM:WAVeform:PHASe` on page 321
`IPM:WAVeform:COUNt` on page 320
`IPM:WAVeform:PERiod` on page 321

Interpolated Shape

The IPM shape is a sequence of values defined in a list, where the list items are mapped to a number of repetitions ("Pulse Count") or equally distributed over a selected period of time ("Period Time").

The transition between the increments can also be linearly interpolated.

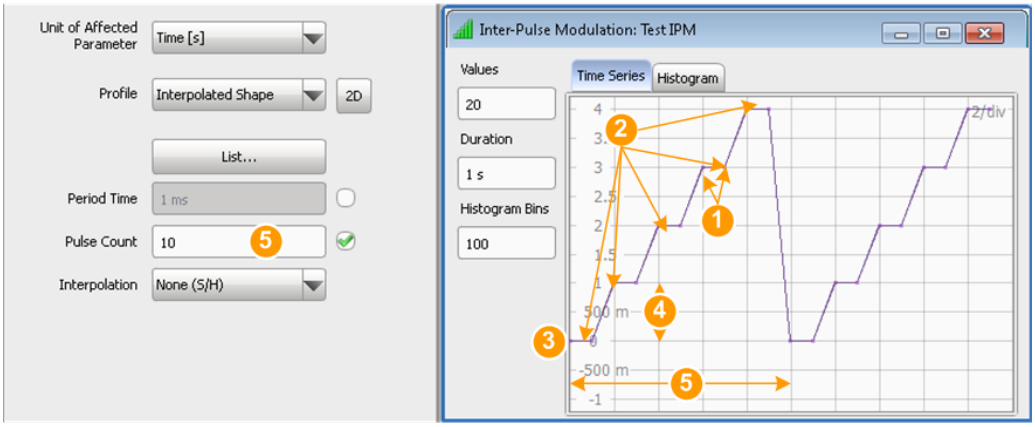


Fig. 9-6: IPM with shape "Profile = Interpolated Shape" ("Interpolation = None (S/H)")

1 = Each increment is repeated two times = "Pulse Count"/"List Count"
2 = "List Count" = 5
3 = "List Start" = 0
4 = "List Increment" = 1
5 = "Pulse Count" = 10

"Period Time" Sets the period of time over that the list items are equally distributed.

"Pulse Count" Sets the number of pulses for that the data from the list is used.

Remote command:
`IPM:SHAPe:BASE` on page 317
`IPM:SHAPe:PERiod` on page 318
`IPM:SHAPe:COUNt` on page 317
`IPM:SHAPe:INTerpol` on page 318

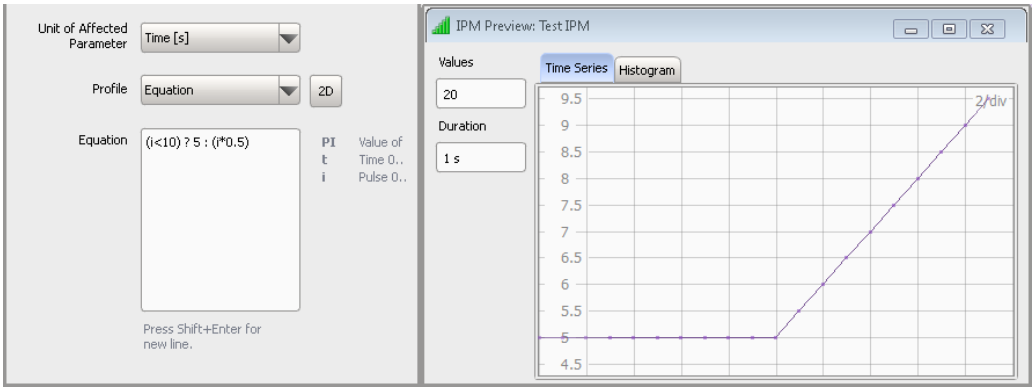
Equation
Defines the IPM shape as a function of:

`t = 0 .. T` is the absolute time

`i = 0 .. N-1` is a pulse counter, where N is the maximum number of pulses

Example:

(i<10) ? 5 : (i*0.5)



See chapter A.2, "Formula Syntax", on page 398.

Remote command:
`IPM:EQUation` on page 314

Random List
The IPM shape is a sequence of discrete values that are randomly selected from a list with user defined values.

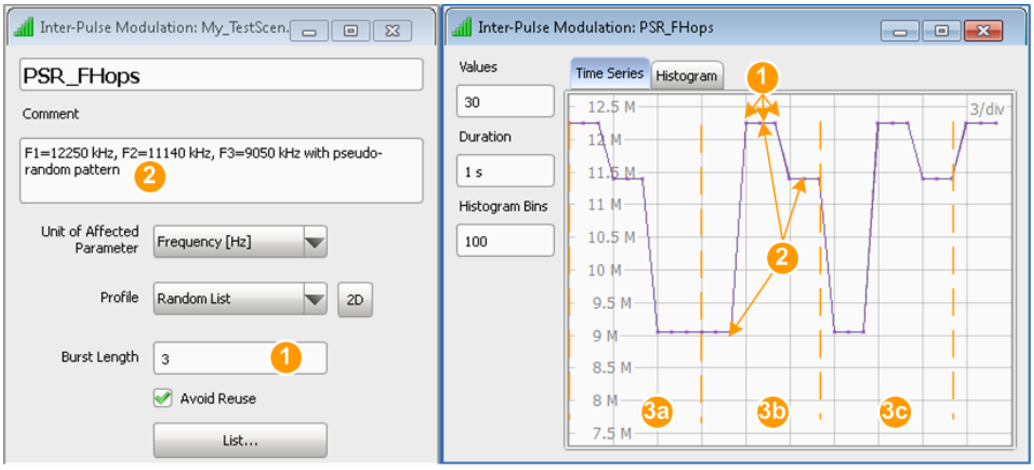


Fig. 9-7: IPM with shape "Profile = Random List"

1 = "Burst Length" = 3
2 = "List Items" = 3
3a, 3b, 3c = Subsequent patterns with values that are selected on a pseudo-random manner from the defined 3 list items

"Burst Length" Defines how often an increment is repeated.

"Avoid Reuse" Ensures that each value is used only once per burst.

"List" Accesses the standard list editor to define a list in table form, see "Edit List" on page 140

- To create a list manually, use the standard icons and functions in the context menu (see "Standard function in the context menus" on page 24). Enter the list values manually.
- To import a list in ASCII format, select "Import Data from file" icon, navigate to a suitable file, select it and import it.
- To create a list automatically and fill it in with values, select the "Populate List" icon.
  Enter the "Start" and "Increment" values and the number of list elements ("Count") and select "Populate".

Remote command:
`IPM:RLISt:BURSt` on page 317
`IPM:RLISt:REUSe` on page 317

Random Steps
The IPM shape is a sequence of desecrate values and step size that are randomly selected from a user defined value ranges.

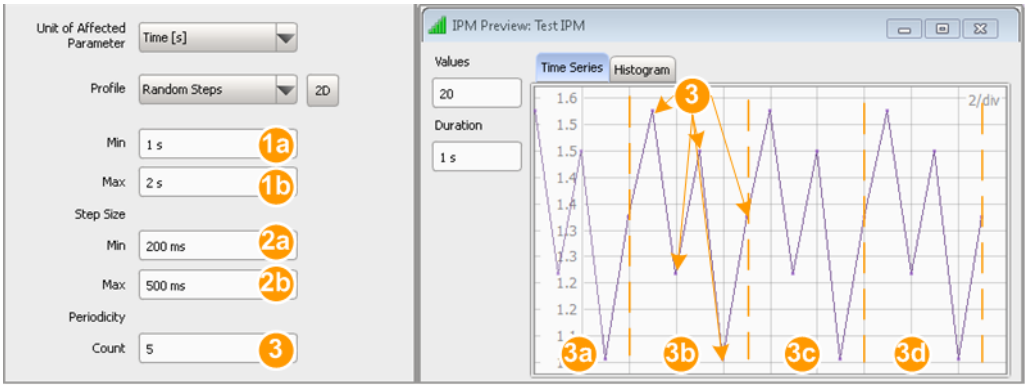


Fig. 9-8: IPM with shape "Profile = Random Steps"

1a, 1b = User defined value range
2a, 2b = User defined step size
3 = "Periodicity > Count" = 5, i.e. selected are 5 random values
3a .. 3d = Subsequent patterns

"Periodicity > Count"
Sets the pattern length, i.e. the number of values that are selected on a pseudo-random manner

Remote command:
`IPM:RSTep:MINimum` on page 319
`IPM:RSTep:MAXimum` on page 319
`IPM:RSTep:STEP:MINimum` on page 319
`IPM:RSTep:STEP:MAXimum` on page 319
`IPM:RSTep:PERiod` on page 319

Random

Random jitter is unlimited in its value range and are usually caused by thermal noise and similar effects. The IPM shape is a sequence of random values according to the select "Distribution" function.

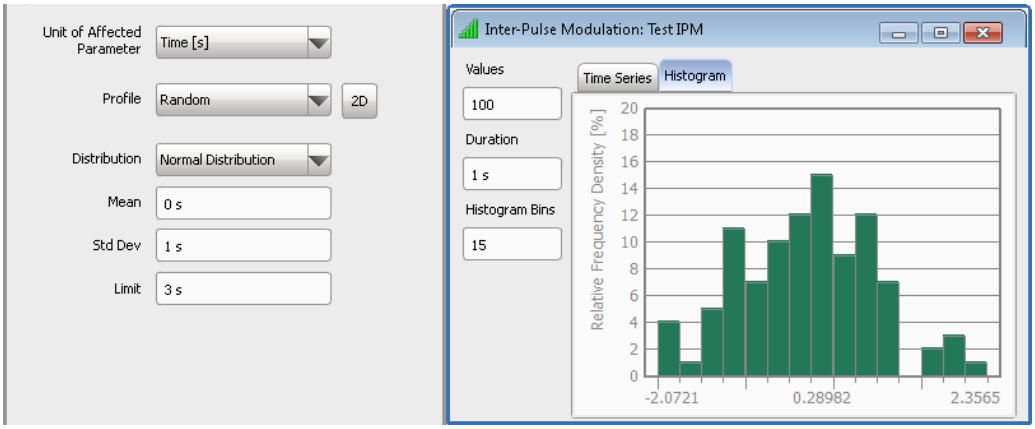


Fig. 9-9: IPM with shape "Profile = Random" and "Distribution = Normal Distribution"

"Uniform Distribution"

With this distribution values occur with the same probability in the range between the minimum and maximum level. The granularity is the "Step" value.

"Normal Distribution"

The Gauss probability density function or normal distribution is characterized by the static parameters "Mean" (μ), "Standard Deviation" (σ) and "Limit".

The figure 9-10 illustrates the probability at which values would occur related to the standard deviation if no limit was set.

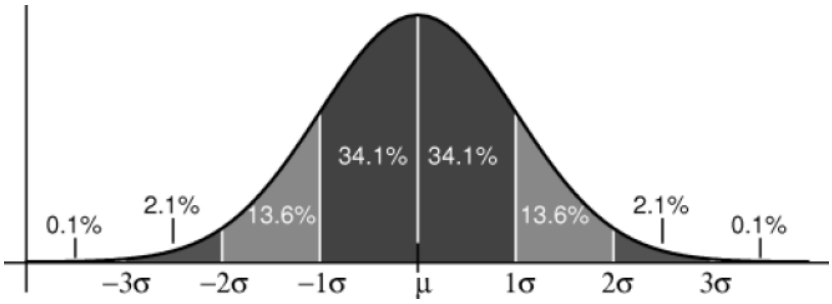
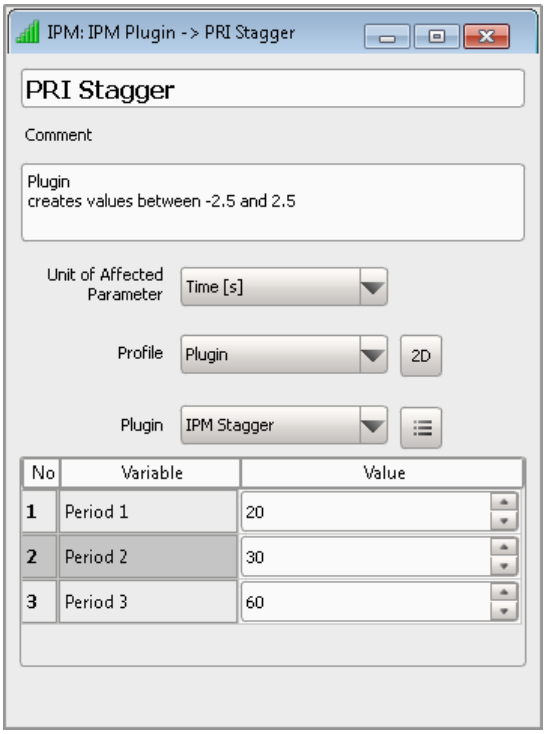


Fig. 9-10: Normal distribution

"U Distribution" The U distribution is characterized by a function with lower limit and upper limit that are calculated from selected "Center" and "Range" values.

Remote command:
`IPM:RANDom:DISTribution` on page 315
`IPM:RANDom:UNIForm:MINimum` on page 316
`IPM:RANDom:UNIForm:MAXimum` on page 316
`IPM:RANDom:UNIForm:STEP` on page 317
`IPM:RANDom:NORMal:MEAN` on page 315
`IPM:RANDom:NORMal:STD` on page 316
`IPM:RANDom:NORMal:LIMit` on page 315
`IPM:RANDom:U:CENTer` on page 316
`IPM:RANDom:U:RANGe` on page 316

Plugin
Custom IPM profiles are used in the same way as the internal IPM profiles.



See chapter 16, "Defining Complex Modulation Schemes and IPM Profiles", on page 232.

"Plugin Name" Selects and loads a user-defined IPM profile from a file.
The IPM profile must exist in the "Plugin" library of the repository.

"Plugin Variables"

Sets the values of the variables defined and used in the plugin.

Use the standard context menu functions to:

- reset the plugin variables
- access the load "Plugin" settings, see chapter 16, "Defining Complex Modulation Schemes and IPM Profiles", on page 232

Remote command:

`IPM:PLUGin:NAME` on page 278

`IPM:PLUGin:VARiable:CATalog?` on page 329

`IPM:PLUGin:VARiable:SELect` on page 329

`IPM:PLUGin:VARiable:VALue` on page 329

9.1.3 Edit List Settings

Edit List

List editor with standard functions.



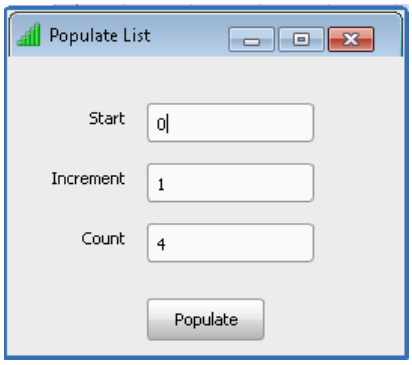
1 = Standard functions for item handling
2 = "Import Data from File" icon
3 = "Populate list" icon

"New/Select" Accesses standard functions for item handling like "Select All", "Remove All", "Insert After", etc.
See table 3-4.

"Import Data from File"

Opens the Windows Explorer to navigate and select an ASCII file with list data.

"Populate List" Accesses the "Populate List" dialog.



To fill in a list with values automatically, set:

- "Start": start value
- "Increment": step with that the values are incremented
- "Count": number of list items.

"Item" Selects an item.

"No" Automatically assigned number to each list item.

"Value" Enters the value of the list item.

"Repetitions" Defines how many times a particular list item is repeated.

Remote command:

`IPM:LIST:LOAD` on page 314

`IPM:LIST:SAVE` on page 315

`IPM:LIST:CLEar` on page 282

9.1.4 Configure Inter-Pulse Modulation (IPM) Settings

To access these settings:

► In the "Sequence" dialog, select a pulse item and select "IPM > Static".

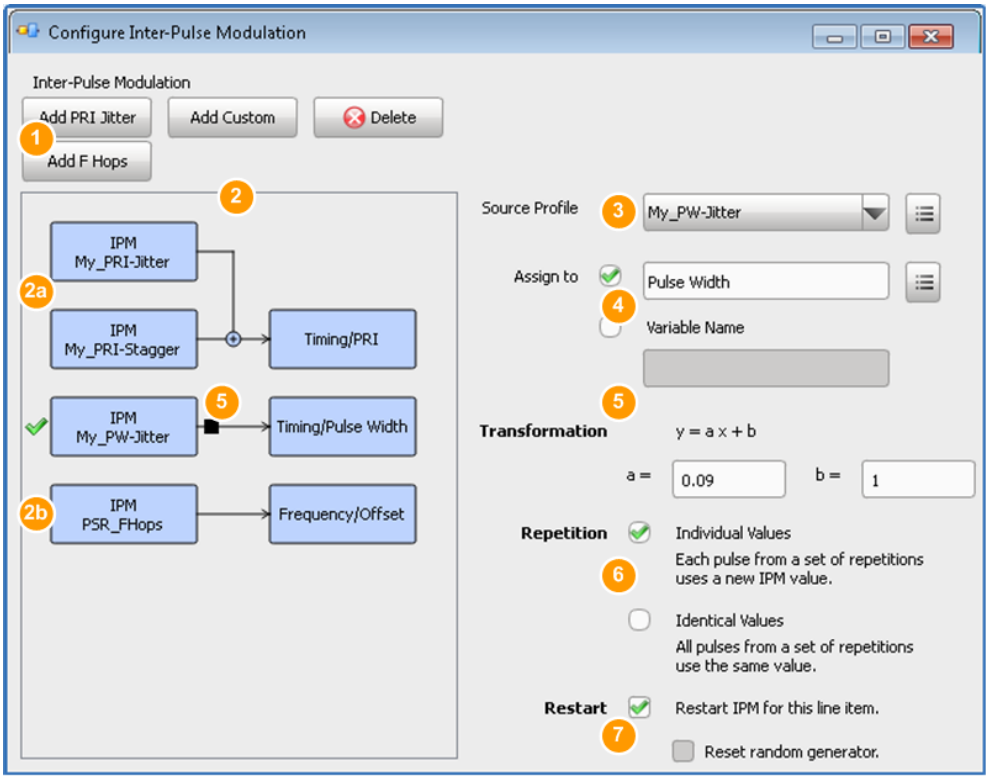


Fig. 9-11: Configure Inter-Pulse Modulation: Understanding the displayed information

1 = Standard add/delete functions to create new IPM of the selected type or remove the selected IPM from the repository
2 = Graphical representation of the configured variations and the parameters they are applied on
2a = Two variations are assigned the same element ("Assign to > Timing > PRI"); these variations are added
2b = Second variation is assigned to a different element ("Assign to > Frequency > Offset")
3 = Variation profile
4 = Defines the pulse parameter or the variable the IPM is applied to
5 = Enables a transformation
6 = Sets the way repeating pulses are processed
7 = Configures the random generator

Add PRI Jitter/Add Frequency Hoping/Add Custom/Delete 143
IPM Graph 143
Source Profile 143
Assign to 143
Transformation 143
Repetition 143
Restart 144

Add PRI Jitter/Add Frequency Hoping/Add Custom/Delete
Standard functions to create new IPM of the selected type, to disable the selected IPM or to remove an automatically created one from the repository.

Remote command:
`SEQuence:ITEM:IPM:ADD` on page 279
`SEQuence:ITEM:IPM:COUNt?` on page 280
`SEQuence:ITEM:IPM:SELect` on page 280
`SEQuence:ITEM:IPM:DELete` on page 281

IPM Graph
Graphical representation of the configured variations and the elements they are applied on.

Source Profile
Lists the available Jitter and IPM profiles in the repository.

To create a new profile or to edit an existing one, select the "New/Select" icon.

Remote command:
`SEQuence:ITEM:IPM:SOURce` on page 322

Assign to
Profiles can be assigned to an existing pulse parameter or to a variable.

- To select a parameter, use the "New/Select" icon
- To select a variable, enter the variable name. This variable will be available within the current sequence segment (table row).

Parameter variations can be applied on the pulse parameters listed in table 9-1.

Remote command:
`SEQuence:ITEM:IPM:TARGet:TYPE` on page 313
`SEQuence:ITEM:IPM:TARGet:PARameter` on page 313
`SEQuence:ITEM:IPM:TARGet:VARiable` on page 314

Transformation
Enables an optional mathematically defined transformation.

The transformation is defined by the equation `y = a*x+b`, where:

- the multiplier "a" and the offset "b" are user defined
- x is the raw value from the IPM profile
- y is the output value of the IPM

Remote command:
`SEQuence:ITEM:IPM:A` on page 321
`SEQuence:ITEM:IPM:B` on page 321

Repetition
Defines the way the variations are applied on repeating pulses:

"Individual value"
A new value is calculated per pulse.

"Identical value"
The same values are used for all repetition of a particular item in the Sequence > Sequence Description Table.
New values are selected each time the same IPM profile is applied to a different item.
Use the "Restart" options to reset an IPM profile.

Remote command:
`SEQuence:ITEM:IPM:MODE` on page 322

Restart
If the IPM profile uses a random generator, you can:
- Restart the IPM per sequence line item, i.e.:
  – resets incremented values
  – starts processing of list from the first item
  – restarts plugins
- Reset the start seed of random generator to reproduce the generated random sequence at a latter time.

Remote command:
`SEQuence:ITEM:IPM:RESTart` on page 322
`SEQuence:ITEM:IPM:RANDom:RESet` on page 322

9.2 How to Create IPM Profiles and Use Them to Vary Pulse Parameters

See:
- "To access the IPM profile dialog" on page 144
- "To create a simple staggered PRI IPM profile" on page 145
- "To assign the staggered PRI profile to the PRI parameter of a pulse" on page 146
- "To visualize the impact of the IPM profiles" on page 148
- "To create and assign a PRI Jitter" on page 151
- "To vary pulse parameters simultaneously" on page 152
- "To create and apply an F Hop (frequency hopping) pattern" on page 149

To access the IPM profile dialog

Perform one of the following:

1. Select "Repository Tree > IPM > New".
2. Open the "Repository Tree > Sequence > Sequence Description > IPM > IPM/ Static > Configure Inter-Pulse Modulation" dialog and:
   a) on the right of "Source Profile" parameter, open the context menu
   b) select "New IPM".

   A new empty IPM profile is created and added to the project tree.
   New profiles are named `IPM <n>` where n is a number starting at one.

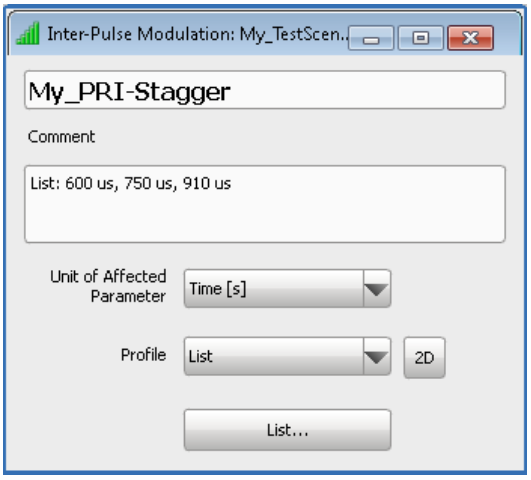
You can add information to describe and identify the profile, like a name or a comment.

To create a simple staggered PRI IPM profile

In this example, we will create an IPM profile that contains three time values, defined in a list form. We will then assign this IPM profile to the PRI parameter of a pulse, see "To assign the staggered PRI profile to the PRI parameter of a pulse" on page 146.

Generated is a pulse train containing three pulses, as for example the pulse train on figure 9-2.

1. Open an IPM profile dialog.
   See "To access the IPM profile dialog" on page 144.
2. Select "Unit >Time [s]" and "Profile > List"



   See also chapter 9.1.2, "IPM Profiles Settings", on page 132.
3. Select "List..." and create a list with three items. The value of each item is set to the required PRI; items are processed once.

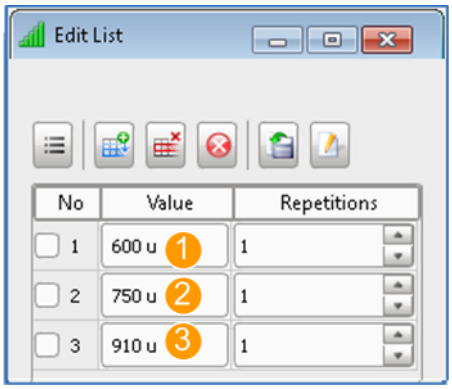
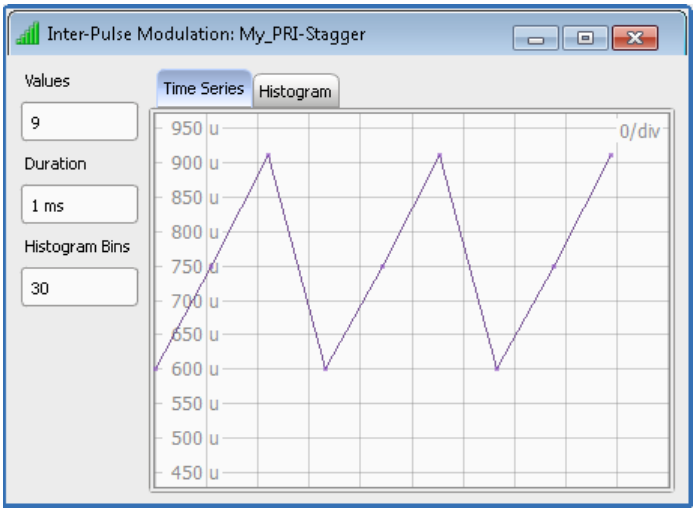


Fig. 9-12: Example of a simple staggered PRI IPM profile

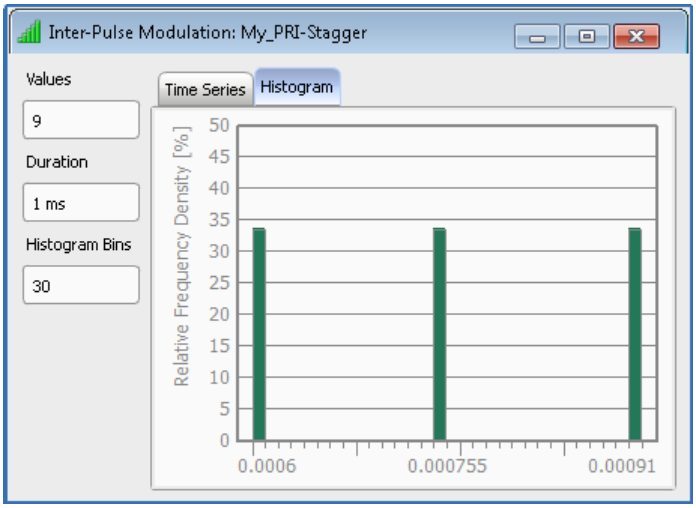
1 = PRI_1 = 600 us
2 = PRI_2 = 750 us
3 = PRI_3 = 910 us

4. In the "IPM" dialog, select "2D > Time Series" to visualize the configured IPM profile in the time domain.



Created is a simple profile with three values only.

5. Select "Histogram" to retrieve statistical information on the IPM profile.



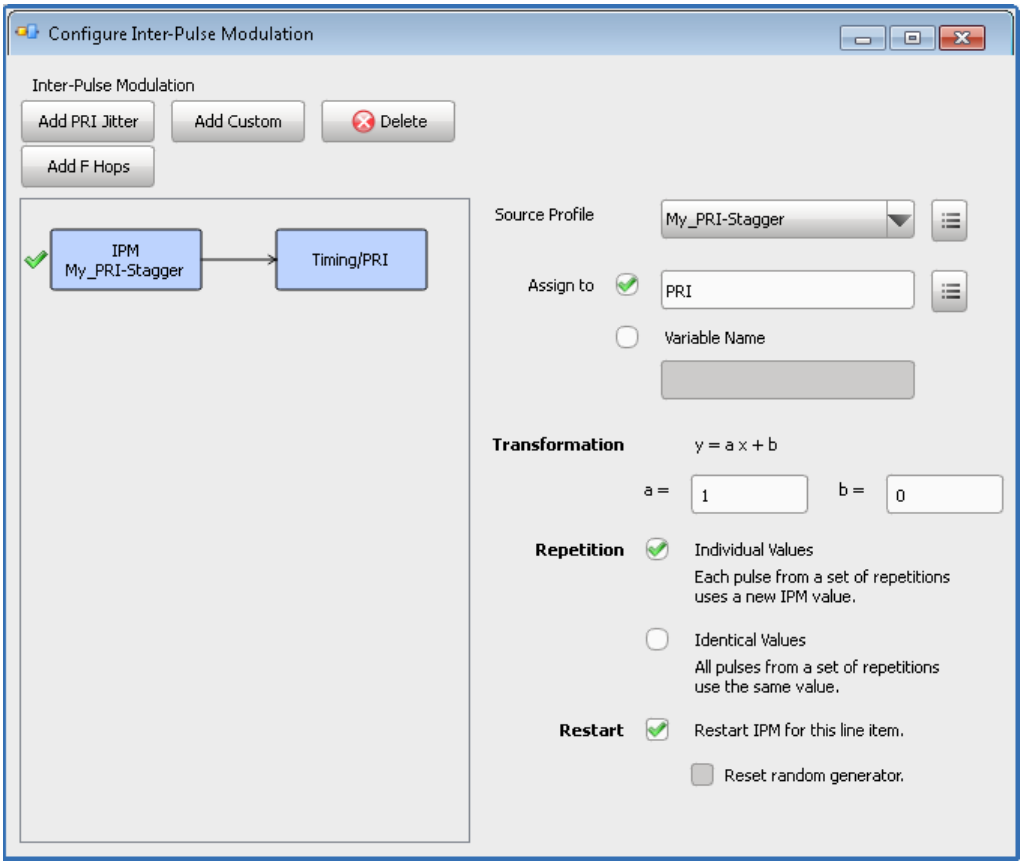
In practice, an staggered PRI profile will follow more complex structure and comprise of a larger number of pulses, see for example figure 9-4.

To assign the staggered PRI profile to the PRI parameter of a pulse

1. Open the pulse sequence the created profile should be applied on, e.g. "My_PT_PRI-Stagger".
   See also "To create a simple sequence" on page 117.
2. In the "Sequence" dialog, select a pulse item and select "IPM > Static".

   The "Configure Inter-Pulse Modulation" dialog opens.
   Per default, there are no predefined parameter variations.
3. To assign the created IPM profile:

a) Select "Add Custom".
New empty profile "IPM NONE" is created.
b) Select "Source Profile > My_PRI_Stagger".
c) On the right of "Assign to" parameter, open the context menu and select "Timing > PRI".



For description of the provided settings, see chapter 9.1.4, "Configure Inter-Pulse Modulation (IPM) Settings", on page 142.

4. In the "Sequence" dialog:

a) set "Repetition Count = 12"
b) set "Delay = 230 us" to enable a stagger start delay, see figure 9-2



The "Sequence" dialog confirms, that a parameter variation varying the PRI is applied ("IPM = IPM" and "PRI = var").

To visualize the impact of the IPM profiles

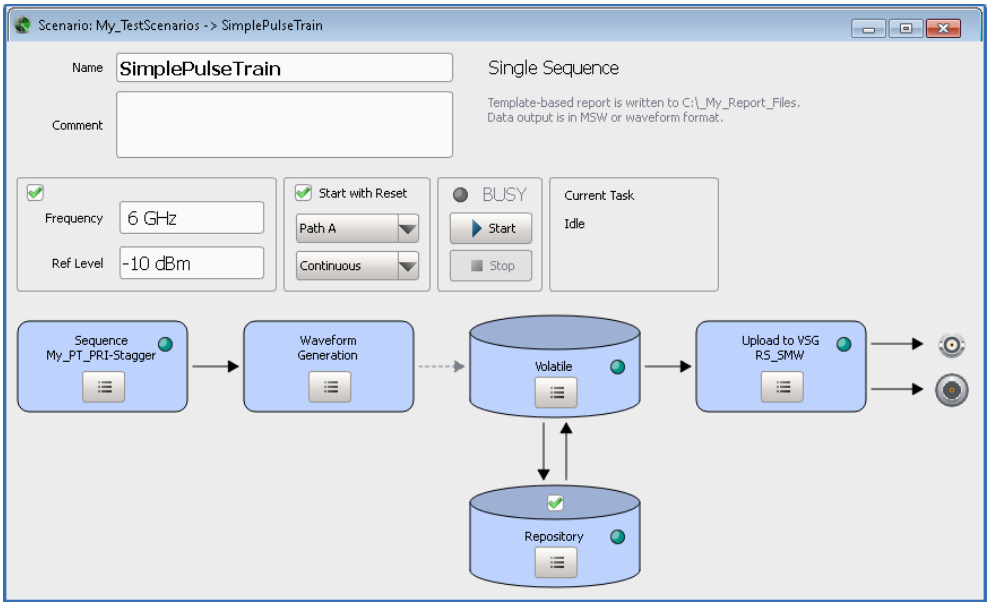
We assume, that a PRI Stagger has been configured and applied to a pulse train, as described in "To assign the staggered PRI profile to the PRI parameter of a pulse" on page 146.

We also assume, that a "Pulse Train" or a "Pulse Train (Collection)" scenario is configured, a signal generator is selected and the sequence is assigned to this scenario; see chapter 6.3, "How to Select and Create a Test Scenario", on page 63.

To visualize the resulting sequence:

1. In the "Scenario" dialog, select "Start".

   The software calculates the waveform. Green LEDs indicate that the processing is completed.



2. Select "Volatile > View".

   The "Waveform Viewer" confirms the configured pulse sequence with enabled PRI Stagger.

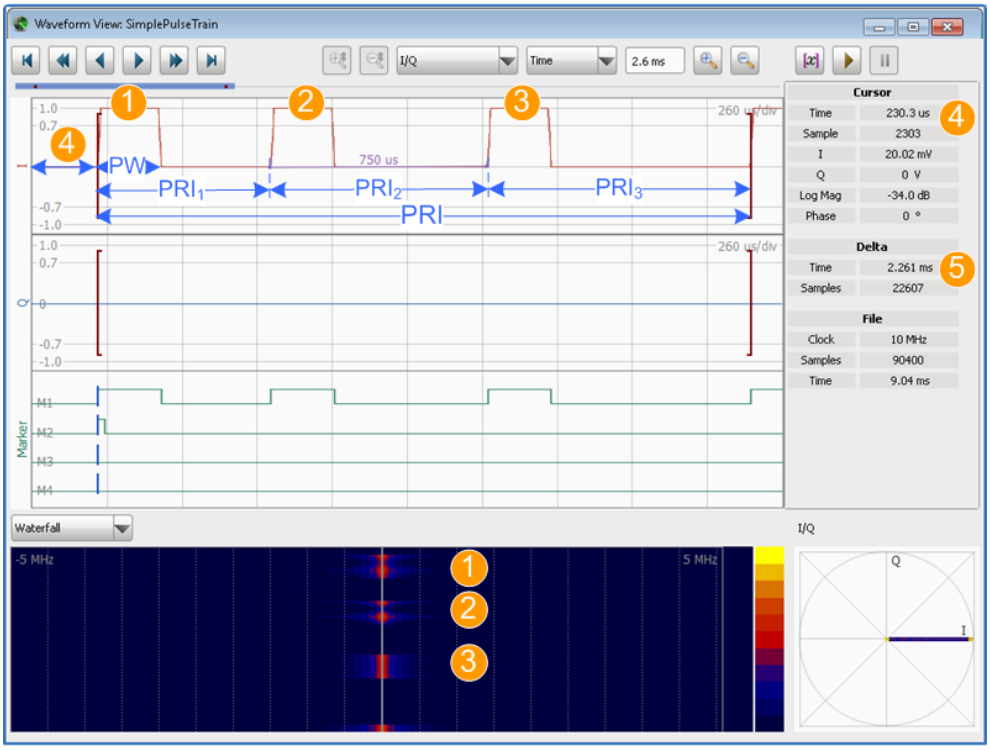


Fig. 9-13: Waveform Viewer: Example of a pulse train (PW = 200 us, PRI stagger)

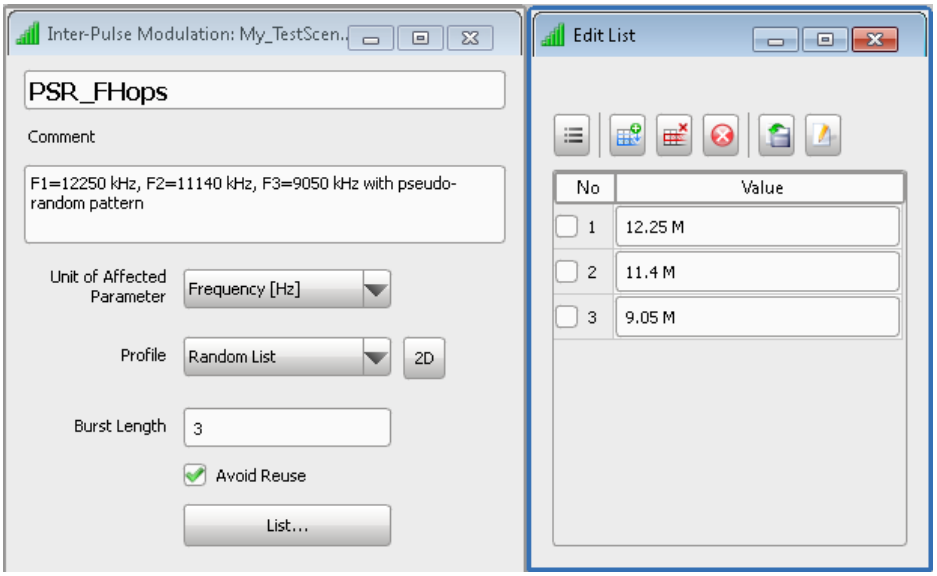
1, 2, 3 = Three unmodulated pulses, also observed on the Waterfall diagram
4 = "Delay = 230 ms"
PRI_1 = 600 us
PRI_2 = 750 us
PRI_3 = 910 us
5 = PRI = 2260 us

For detailed description of the provided settings, see chapter 18.1, "Waveform and Data View Settings", on page 240.

To create and apply an F Hop (frequency hopping) pattern

1. In the "Sequence" dialog, select IPM to open the "Configure Inter-Pulse Modulation" dialog.
2. Select "Add F Hops".

   A new IPM profile is created, automatically named and assigned to the "Frequency/ Offset" parameter.
3. Configure the parameters for example as shown on the following figure.



4. Use the 2D view to display the configured pattern, see also figure 9-7.
5. Observe the "Sequence" dialog.

   The sequence configuration settings confirm the configuration.



6. Use the "Waveform Viewer" to visualize the generated waveform.

   See also:

   - "To visualize the impact of the IPM profiles" on page 148

- chapter 18.1, "Waveform and Data View Settings", on page 240

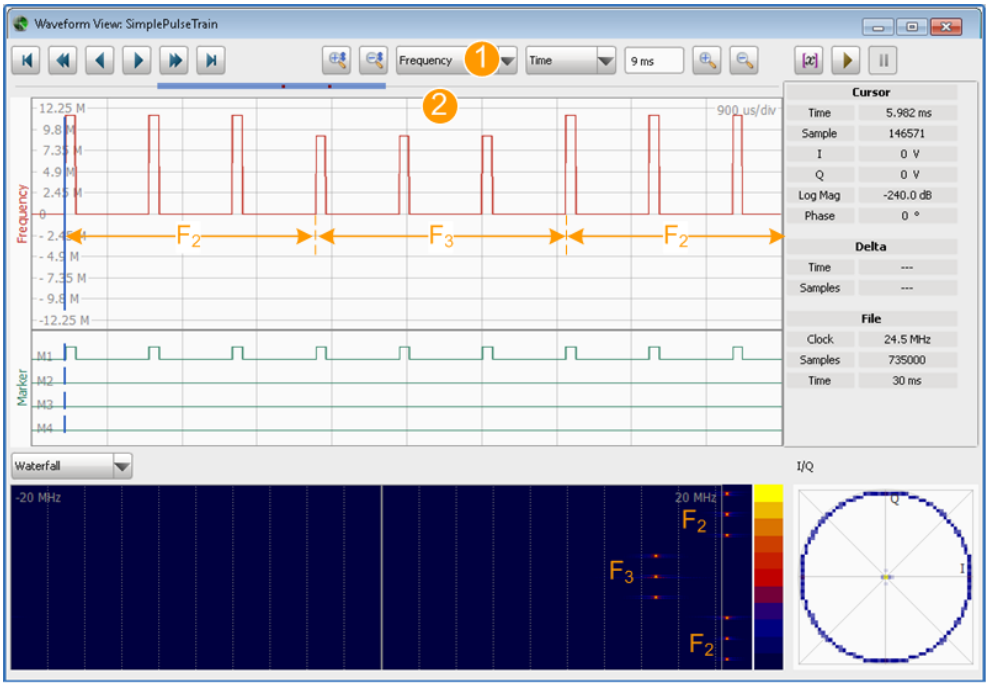


Fig. 9-14: Waveform Viewer: Example of a pulse train (PW = 100 us, Frequency hopping)

1 = "View Mode > Frequency"
2 = 30 standard pulses; 3 consecutive pulses per hopping frequency
F1 = 12250 kHz (not on the display)
F2 = 11140 kHz
F3 = 9050 kHz

The signal will hop the three frequencies according to a pseudo-random pattern. Each hop frequency is used three times.

To create and assign a PRI Jitter

There is a dedicated function for creating PRI Jitters, the "PRI Jitter" in the "Configure IPM" dialog (see also "To create and apply an F Hop (frequency hopping) pattern" on page 149).

In this example, however, we create manually a randomly varying IPM profile with 20 different values. Within 20 ms, the PRI should vary in the range between 1.1 ms and 1.15 ms.

1. Create a new IPM Profile that uses "Profile > Random" and for example a "Distribution > Uniform Distribution".
   See also "To access the IPM profile dialog" on page 144.



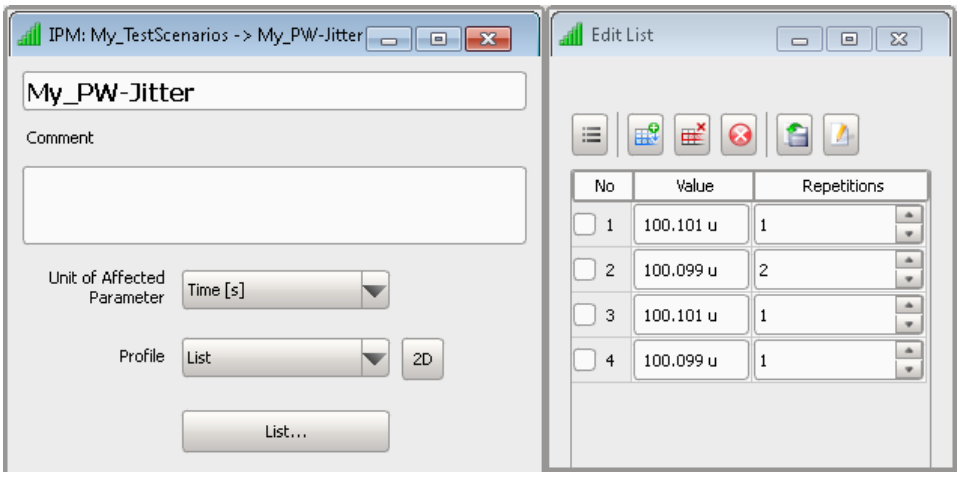
2. Select "2D" to visualize the function.
3. Open the required sequence and assign the IPM profile to a pulse train.
   See "To create and apply an F Hop (frequency hopping) pattern" on page 149.

To vary pulse parameters simultaneously

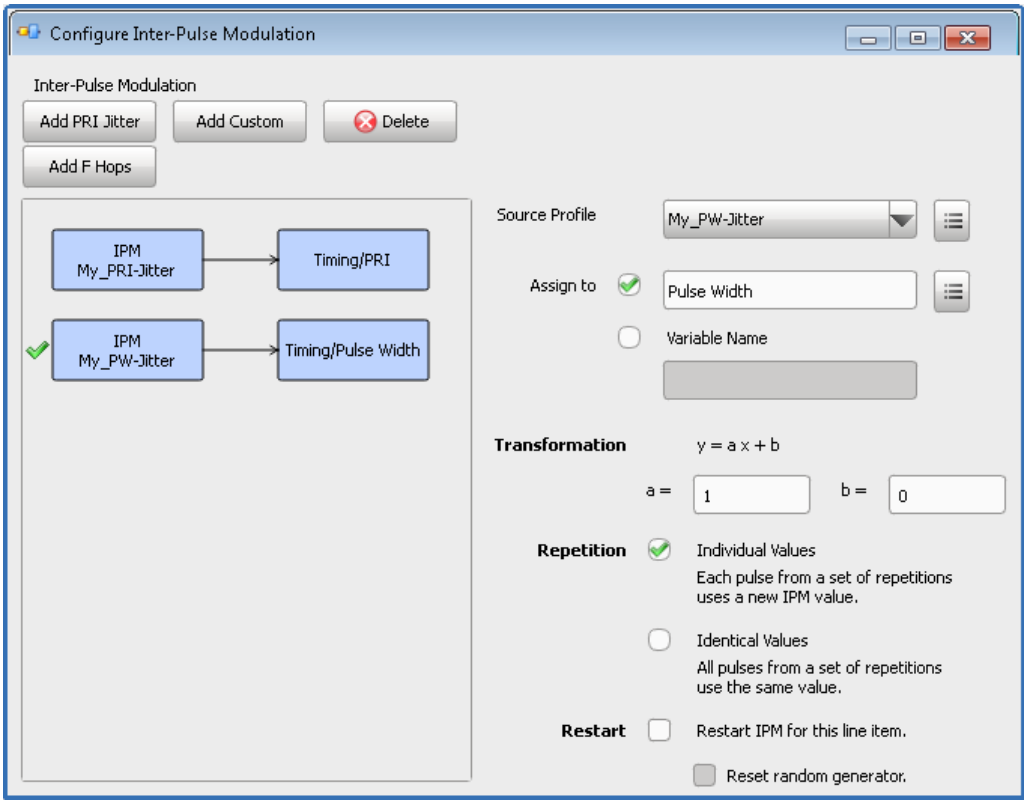
We assume, that a PRI Jitter has been configured and applied to a pulse train, as described in "To create and assign a PRI Jitter" on page 151.

To apply a second variation, for example a PW Jitter, perform the following:

1. Create a new IPM Profile, for example as shown on the following figure.



   Configured is a pulse width (PW) pattern with 5 values.
2. Open the required sequence and assign the IPM profile to the same pulse train as the PRI Jitter.



Tip: You can vary one pulse parameter with more than one profiles, see figure 9-11.

Configured is a pulse train, consisting of 20 pulses, each with different PRI value; the PW pattern is applied on 5 consequent pulses and is repeated 4 times.
In this example, the PRI varies in the range 1.1 ms to 1.15 ms; the PW between 100.101 us and 100.099 us.

10 Defining Antenna Patterns and Antenna Scans

This section informs you about the provided antenna pattern types and antenna scan types. It also helps you select and configure a subset of antenna characteristics as basis for the emitter configuration and its different operating modes.

10.1 Overview of the Main Antenna Parameters

Antennas are elements that radiate energy. Radiated energy focused in one defined direction is referred to as a main beam or main lobe (see figure 10-1). Radiated energy in other than the required direction form the side lobes and the back lobe.

The angle at that the radiated energy falls down to the half of the maximum is referred to as half-power beam width (HPBW).

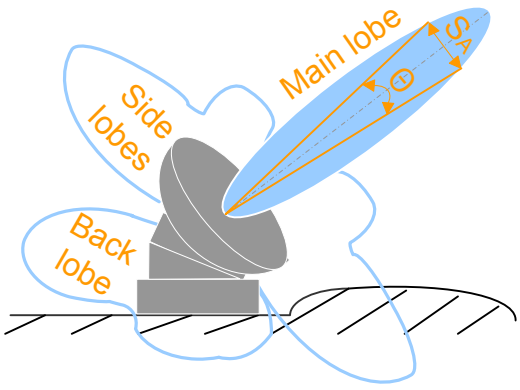


Fig. 10-1: Main characteristics of an antenna

Main lobe/bream = The region around the direction with maximum radiation
Side lobes = Energy radiated in undesired direction
S_A = Angular resolution
Θ (Theta) = Antenna beam width or the -3dB angle, defined by the half-power (HPBW)

Antenna pattern

Antennas of different type and with different purpose use more wider or more directed beams, for example pencil beams have a typical HPBW of about 2 deg.

The R&S Pulse Sequencer provides a wide range of commonly used antenna patterns, such as Parabolic, Cosecant Squared, or Planar Phased Array antennas. Moreover you can:

- adjust the antenna pattern settings to simulate for example a directive antenna that concentrates the radiated energy in a particular direction

- simulate a back lobe pattern
- create a custom antenna pattern based on the required antenna characteristics
  see "To create a custom antenna pattern based upon the required antenna characteristics" on page 175
- import your user-defined antenna pattern
  see "To import a user-defined antenna pattern from file" on page 176
- visualize the antenna pattern on a 2D and a 3D preview, see figure 10-4.

Antenna scan

Antenna scans describe the movement of the antenna beam in three-dimensional space. A conical scan for example is a movement where the beam points in the direction of the target and is rotated in a circle, i.e. the half-power points of the beam draw a circle.

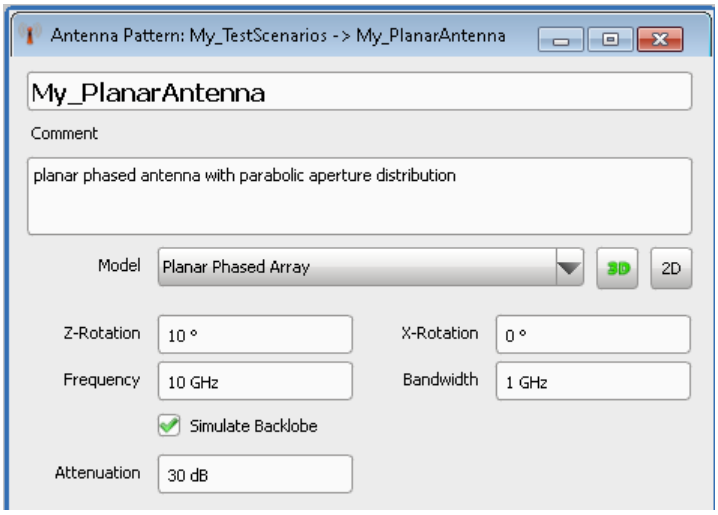
The R&S Pulse Sequencer provides a wide range of commonly used antenna scans, such as a raster scan, helical scan, conical scan etc. You can also:

- adjust the antenna scan settings
- visualize the configured antenna scan on a 3D preview
- observe the signal received by a receiver for a particular antenna pattern and antenna scan configuration
  See:
  – "Receiver" on page 201
  – "To visualize the signal received by a static receiver" on page 188

10.2 Antenna Pattern Settings

To access these settings:

► Select "Repository Tree > Antenna Pattern > New".



Antenna Patterns are characterized with the following common and model specific settings:

Antenna Pattern Name....156
Comment....156
Antenna Model....156
Z-Rotation, X-Rotation....157
Frequency, Bandwidth....157
Simulate Back Lobe, Attenuation, Type....157
2D and 3D diagrams....158
Dipole Antenna Settings....158
Parabolic Antenna Settings....158
Gaussian Antenna Settings....159
Sin(x)/x Antenna Settings....159
Pyramidal Horn Antenna Settings....160
Cosecant Squared Antenna Settings....160
Planar Phased Array Antenna Settings....160
Custom Antenna Settings....161
Imported Antenna Settings....161
Import Wizard....162

Antenna Pattern Name
Enter the name of the antenna pattern.

Remote command:
`ANTenna:NAME` on page 277
`ANTenna:CATalog?` on page 276
`ANTenna:CREate` on page 276
`ANTenna:SELect` on page 277
`ANTenna:REMove` on page 278

Comment
Enter a short description.

Remote command:
`ANTenna:COMMent` on page 278

Antenna Model
Selects the antenna pattern. Use the "2D" and "3D" diagrams to visualize the polar antenna gain patterns.

Further provided antenna pattern settings depends on the selected antenna model, see:

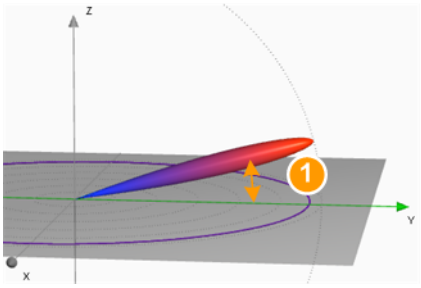
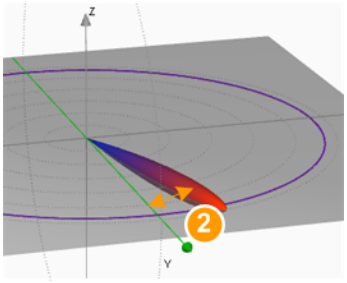
- Dipole Antenna Settings
- Parabolic Antenna Settings
- Gaussian Antenna Settings
- Sin(x)/x Antenna Settings
- Pyramidal Horn Antenna Settings
- Cosecant Squared Antenna Settings
- Planar Phased Array Antenna Settings
- Imported Antenna Settings

- Custom Antenna Settings

Remote command:
ANTenna:MODel:TYPE on page 289

Z-Rotation, X-Rotation
Rotates the antenna pattern around the selected axis.

Table 10-1: Effect of antenna pattern rotation

"X-Rotation = 15°"	"Z-Rotation = 15°"
	
1) Rotation around the x-axis Positive values rotate the antenna boresight towards the positive z-axis.	2) Rotation around the z-axis Positive values rotate the antenna boresight counter clockwise towards the x-axis.

This parameter is useful for example:
- to simulate a radar boresight displacement
- to align a user defined antenna pattern to the boresight;
  see "To import a user-defined antenna pattern from file" on page 176.

Remote command:
ANTenna:MODel:ROTation:X on page 289
ANTenna:MODel:ROTation:Z on page 289

Frequency, Bandwidth
Sets the frequency and bandwidth the antenna pattern is calculated for.

See also:
- Frequency
- Emitter Modes

Remote command:
ANTenna:MODel:FREQuency on page 287
ANTenna:MODel:BANDwidth on page 287

Simulate Back Lobe, Attenuation, Type
Enables the simulation of a back lobe and sets its power level and shape.

The parameter "Type" defines the shape of the back lobe pattern:
- "Mirror": the back lobe pattern is the mirrored antenna pattern, attenuated with the selected value, see figure 10-5.

- "Omnidirectional": the back lobe pattern is a hemisphere, attenuated with the selected value.

Remote command:
ANTenna:MODel:BACKlobe:ENABle on page 285
ANTenna:MODel:BACKlobe:TYPE on page 286
ANTenna:MODel:BACKlobe:ATTenuation on page 286

2D and 3D diagrams
You can visualize the antenna characteristics as 2D and 3D interactive diagrams.

"2D" This diagram is a plot of the radiated energy, measured at different angles and at a constant distance from the antenna. The 2D diagram indicates the calculated HPBW and the 3 dB line.

The 2D diagram display the antenna pattern as a XY or XZ graph with polar or with Cartesian coordinates.

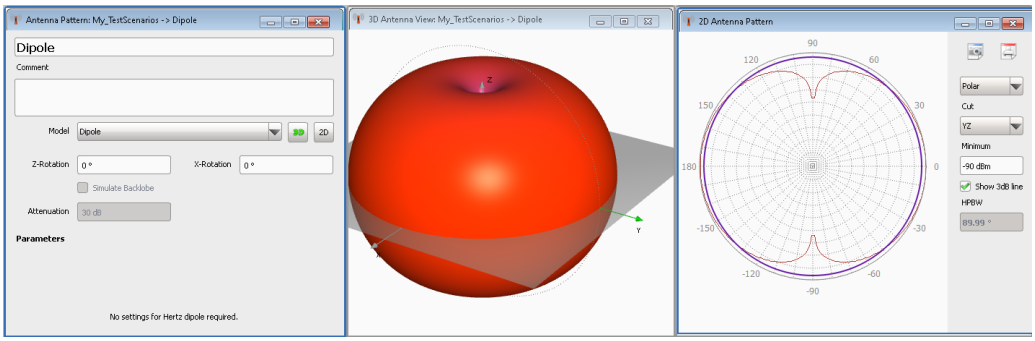
- The polar diagram shows the main, side and back lobes, see figure 10-4.
- The Cartesian diagram is useful to determine the main lobe to back lobe and main lobe to side lobes ratio.

Remote command:
PLOT:POLar:TYPE on page 290
PLOT:POLar:CUT on page 290
PLOT:POLar:LOG:MIN on page 290

"3D" This diagram displays the radiated energy as a 3D model.

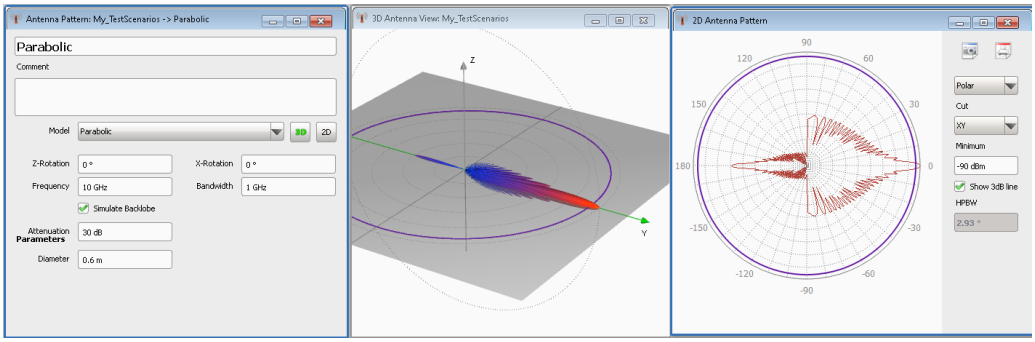
"Minimum Displayed Level"
Defines the minimum displayed side lobes level.

Dipole Antenna Settings
A dipole antenna is a simple antenna with a radiation pattern, shaped like a toroid (doughnut) symmetrical about the axis of the dipole.



Parabolic Antenna Settings
A parabolic antenna has a curved surface with the cross-sectional shape of a parabola with a user defined "Diameter".

Typically, a parabolic antenna radiates the power in a narrow main lobe along the antenna's Y-axis.

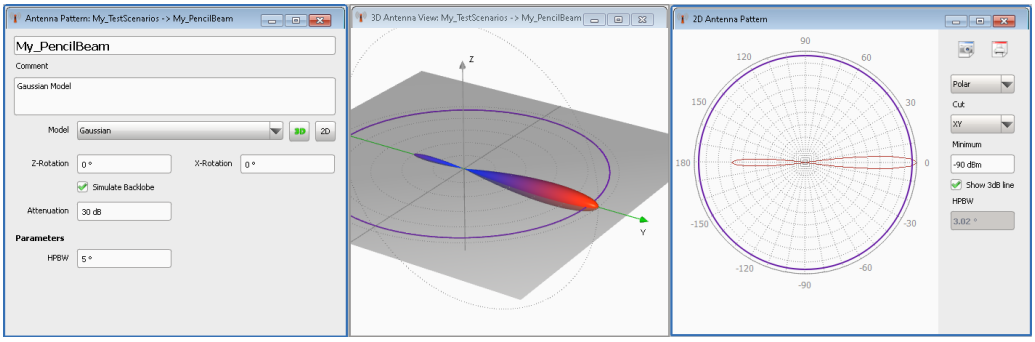


Remote command:
ANTenna:MODel:PARabolic:DIAMeter on page 288

Gaussian Antenna Settings

An antenna with radiation pattern that follows the Gaussian distribution.

The main parameter is the HPBW that specifies the angular width within which the antenna is most sensitive.

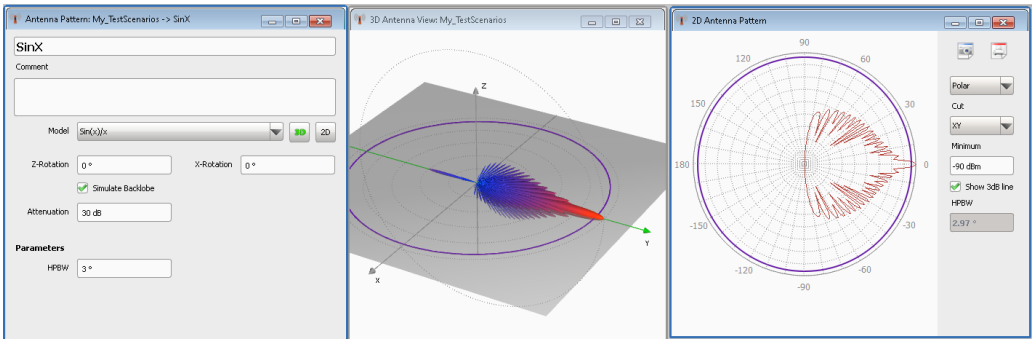


Remote command:
ANTenna:MODel:GAUSsian:HPBW on page 288

Sin(x)/x Antenna Settings

An antenna with radiation pattern that follows the Sin(x)/x distribution.

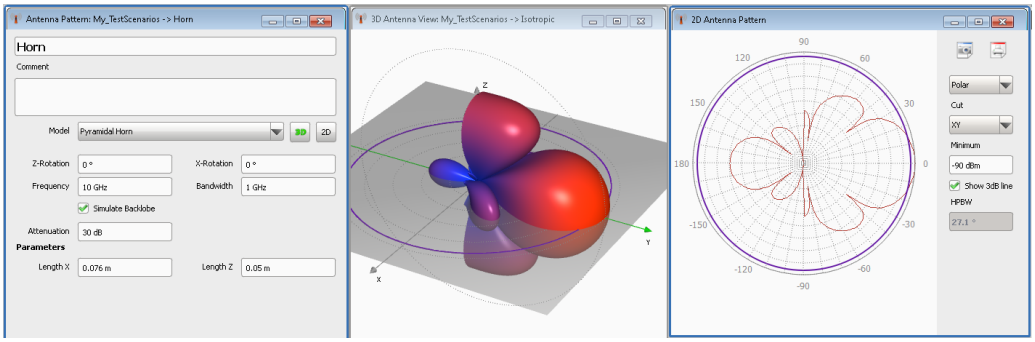
The antenna is characterized by the HPBW.



Remote command:
ANTenna:MODel:SINC:HPBW on page 288

Pyramidal Horn Antenna Settings

A pyramidal horn antenna has a horn shape in the form of a four-sided pyramid and a rectangular cross section. The length of the rectangular sides "Length X, Length Z" are user definable.



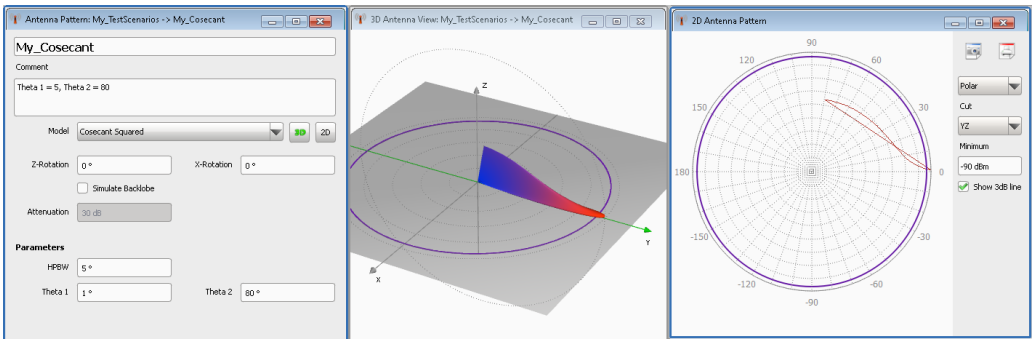
Remote command:

`ANTenna:MODel:HORN:LX` on page 288

`ANTenna:MODel:HORN:LZ` on page 288

Cosecant Squared Antenna Settings

Cosecant squared antennas are designed for air-surveillance radar sets.



Remote command:

`ANTenna:MODel:COSecant:HPBW` on page 288

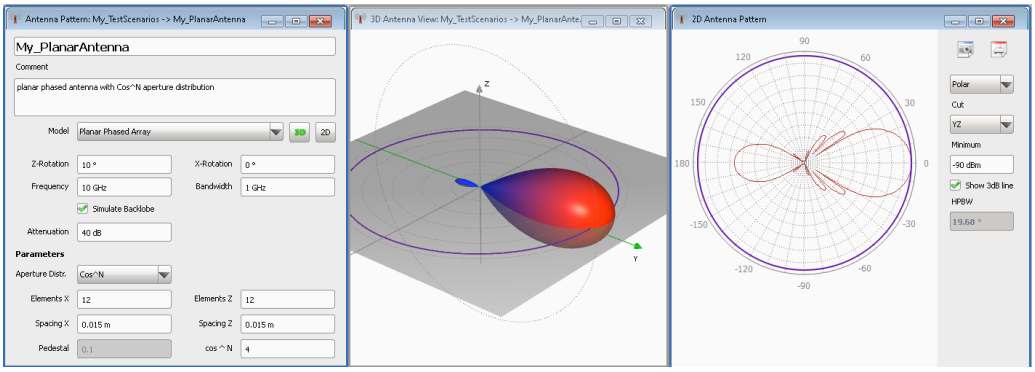
`ANTenna:MODel:COSecant:T1` on page 286

`ANTenna:MODel:COSecant:T2` on page 286

Planar Phased Array Antenna Settings

A planar phased array antenna consists of several antenna elements.

The number of elements and the spacing between them is user-definable ("Elements X/Elements Z, Pedestal"). The exact antenna pattern is defined by the selected "Aperture Distribution".



See also "To create an antenna pattern" on page 171.

Remote command:
ANTenna:MODel:ARRay:DISTribution on page 285
ANTenna:MODel:ARRay:XDIStance on page 285
ANTenna:MODel:ARRay:ZDIStance on page 285
ANTenna:MODel:ARRay:PEDestal on page 285
ANTenna:MODel:ARRay:COSN on page 284
ANTenna:MODel:ARRay:NX on page 285
ANTenna:MODel:ARRay:NZ on page 285

Custom Antenna Settings
You can define an antenna pattern based on the required antenna characteristics.

See figure 10-7.

"HPBW XY, HPBW YZ"
HPBW specifies the angular width within which the antenna is most sensitive.

"Side lobe level"
Attenuation applied on the first pair of side lobes compared to the power level of main lobe

"Roll off factor" Step size, used to calculate the attenuation of the subsequent side lobes

"Side lobe scale"
Scale factor, used to calculate the HPBW of the side lobes, i.e. how much smaller or wider the side lobes are compared to the main lobe

Remote command:
ANTenna:MODel:CUSTom:HPBW:XY on page 286
ANTenna:MODel:CUSTom:HPBW:YZ on page 286
ANTenna:MODel:CUSTom:SLSTart on page 287
ANTenna:MODel:CUSTom:SLRolloff on page 287
ANTenna:MODel:CUSTom:SLSCale on page 287

Imported Antenna Settings
You can define and load you custom antenna patterns.

Use the "Load/Delete" standard functions for file management. You can load your custom antenna pattern file.

The following file formats are supported:

- FEKO Far Field (`*.ffe`) files
- `*.ant_pat` files in the Rohde&Schwarz proprietary format
- Antenna Magus `*.tsv` files
- ANSYS HFSS `*.ffd` files
- Antenna patterns in `*.csv` file format.
  The Import Wizard assists you to import `*.csv` files.

See:

- chapter A.1.2, "Antenna Pattern File Formats", on page 393 for description of the file formats.
- "To import a user-defined antenna pattern from file" on page 176.

Remote command:

`ANTenna:MODel:USER:LOAD` on page 289

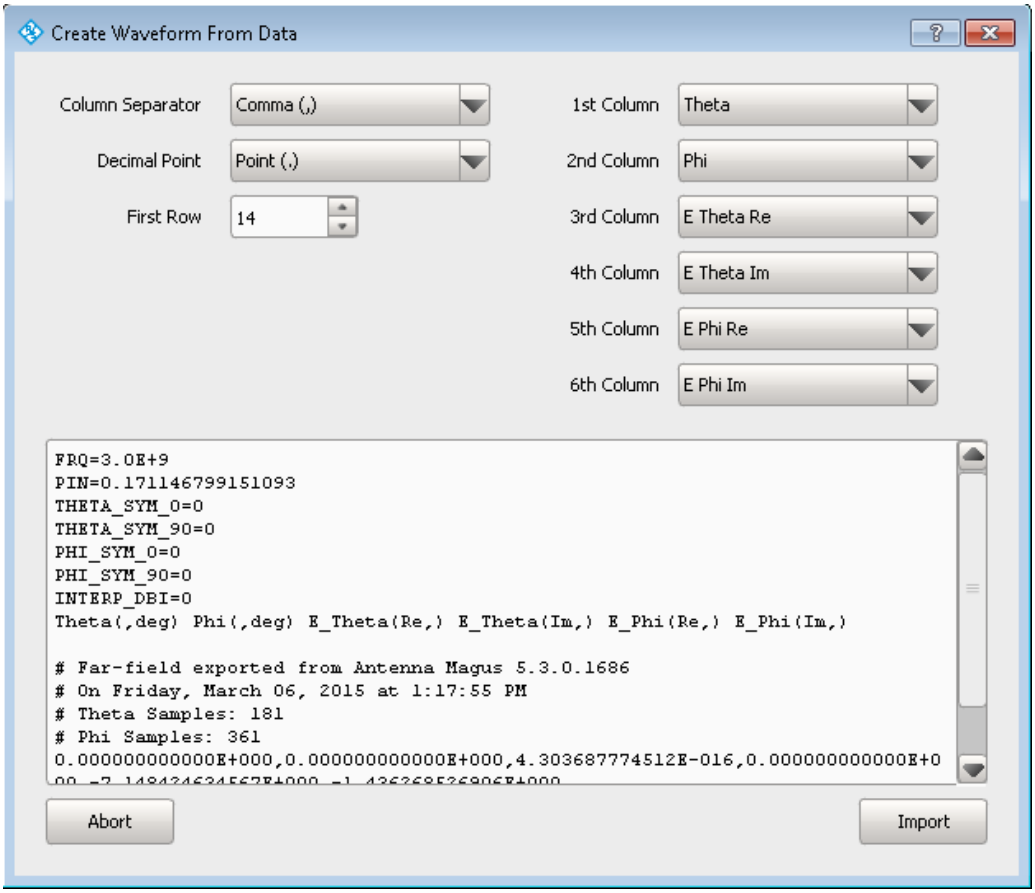
`ANTenna:MODel:USER:CLEar` on page 281

Import Wizard

When a `*.csv` file is loaded, the "Import Wizard" opens and tries to determine:

- the used column separator
- the decimal delimiter
- the first row with data
- the data content of each column.

You can change the values afterwards.

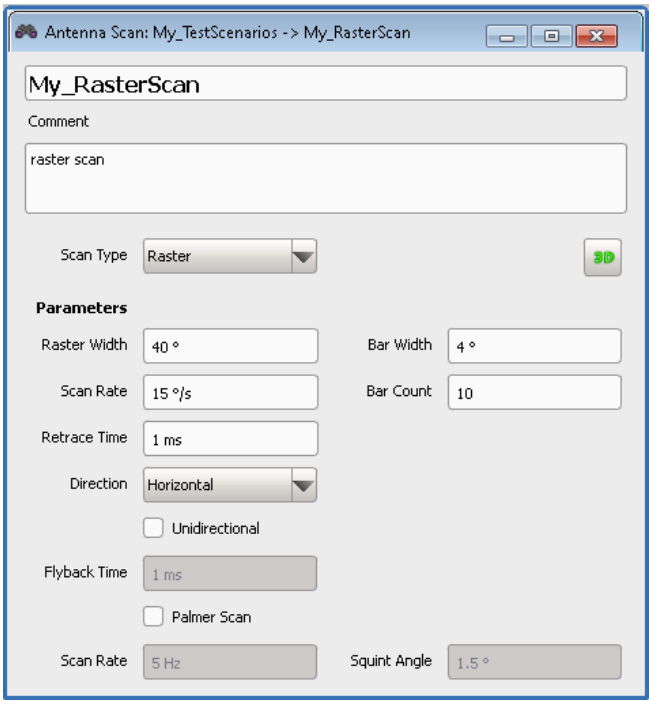


Remote command:
`ANTenna:MODel:USER:CSV:FORMat` on page 289

10.3 Antenna Scans Settings

To access these settings:

► Select "Repository Tree > Antenna Scan > New".



Antenna Scans are characterized with the following settings:

Antenna Scan Name....165
Comment....165
Scan Type....165
3D Scan View....165
└ Scan Animation Time....165
└ Scan and Pattern Visualization....165
Circular Scan Settings....165
Sector Scan Settings....166
Raster Scan Settings....167
Conical Scan Settings....168
Helical Scan Settings....169
Spiral Scan Settings....169
Lobe Switching Scan Settings....170

Antenna Scan Name
Enter the name of the antenna scan.

Remote command:
SCAN:NAME on page 278
SCAN:CATalog? on page 276
SCAN:SELect on page 277
SCAN:CREate on page 277
SCAN:REMove on page 279

Comment
Enter a short description.

Remote command:
SCAN:COMMent on page 278

Scan Type
Defines the scan type.

Use the "3D Scan View" diagram to visualize the selected scan.

Remote command:
SCAN:TYPE on page 297

3D Scan View
Visualizes the antenna scan in a 3D preview.

See the example on figure 10-6.

Scan Animation Time ← 3D Scan View
Sets the time it takes the animation to complete a scan.

The "Real Time" is also displayed.

Scan and Pattern Visualization ← 3D Scan View
Defines the way the scan is represented:

- "Scan Line = On": the scan is represented by a line; the antenna pattern is disregarded.
- "Pattern Visualization > Pattern": the visualization uses a Gaussian pattern.
  If you have opened this dialog form the "Emitter" dialog or the 2D map, the visualization corresponds to the current selected antenna pattern.
- "Pattern Visualization > HPBW": the antenna is represented by its HPBW.
- "Minimum Displayed Level": the minimum displayed side lobes level.

Circular Scan Settings
A circular antenna turns in a circle and scans 360°.

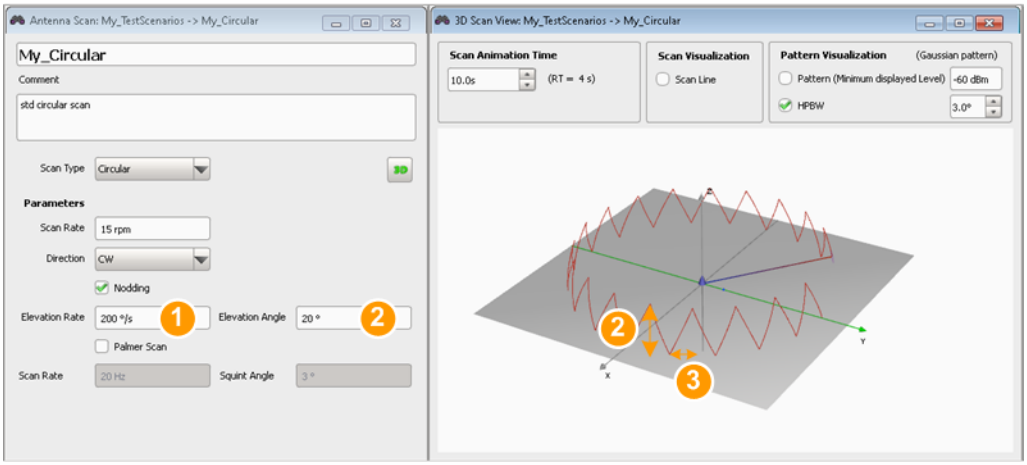


Fig. 10-2: Circular scan with enabled nodding: understanding the displayed information

1 = "Elevation Rate"
2 = "Elevation Angle"
3 = Rise Time = "Elevation Angle"/"Elevation Rate"

"Scan Rate" Sets speed the antenna is turning with.

"Direction" Sets the direction, clockwise ("CW") or counter clockwise ("CCW"), the antenna is turning in.

"Nodding" If enabled, superimposes a horizontal nodding on the circular scan. The height of the nods and their rise time are defined with the parameters "Elevation Angle" and "Elevation Rate", see figure 10-2.

"Palmer Scan" If enabled, superimposes a conical scan on the circular scan, see figure 10-3.

Remote command:
`SCAN:CIRCular:RPM` on page 293
`SCAN:CIRCular:ROTation` on page 292
`SCAN:CIRCular:NODDing` on page 297
`SCAN:CIRCular:NRATe` on page 298
`SCAN:CIRCular:NELevation` on page 297
`SCAN:CIRCular:PALMer` on page 298
`SCAN:CIRCular:PRATe` on page 298
`SCAN:CIRCular:PSQuint` on page 298

Sector Scan Settings
A Sector antenna scans only a sector with defined "Sector Width".

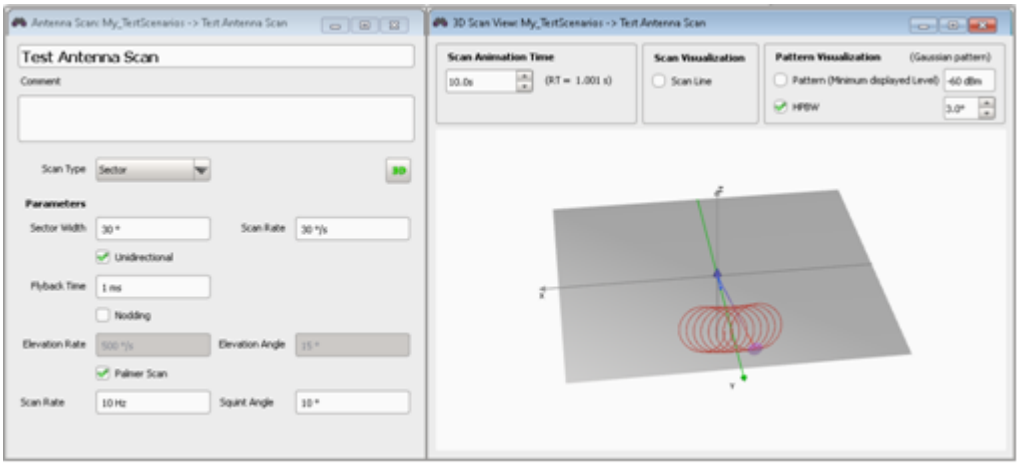


Fig. 10-3: Sector scan with enabled Palmer scan: understanding the displayed information

"Sector Width, deg"
Sets the width of the scanned sector.

"Scan Rate, deg/s"
Defines how fast the antenna is turning.

"Unidirectional" The antenna uses an unidirectional scan mode, i.e. the antenna turns in one direction only and "jumps" back within the selected "Flyback Time".

"Nodding" If enabled, superimposes a horizontal nodding on the circular scan. The height of the nods and their rise time are defined with the parameters "Elevation Angle" and "Elevation Rate", see figure 10-2.

"Palmer Scan" If enabled, superimposes a conical scan on the circular scan, see figure 10-3.

Remote command:
`SCAN:SECTor:WIDTh` on page 296
`SCAN:SECTor:RATE` on page 293
`SCAN:SECTor:UNIDirection` on page 295
`SCAN:SECTor:FLYBack` on page 295
`SCAN:SECTor:NODDing` on page 297
`SCAN:SECTor:NELevation` on page 297
`SCAN:SECTor:NRATe` on page 298
`SCAN:SECTor:PALMer` on page 298
`SCAN:SECTor:PRATe` on page 298
`SCAN:SECTor:PSQuint` on page 298

Raster Scan Settings
The raster antenna scans a sector with defined "Sector Width" but performs several scans with different elevations.

See figure 10-6.

"Raster Width, deg"
Sets the width of the scanned sector.
Scanned is an angle of ±"Raster Width"/2 degrees, starting from the Y axis.

"Scan Rate, deg/s"
Defines how fast the antenna is turning.

"Bar Count, Bar Width"
Sets the number of bars and the distance between two consecutive scanned bars (sectors).

"Retrace Time" Defines how fast the antenna returns to its initial orientation after completing the last bar.

"Direction" Switches between a horizontal and a vertical scanning direction.

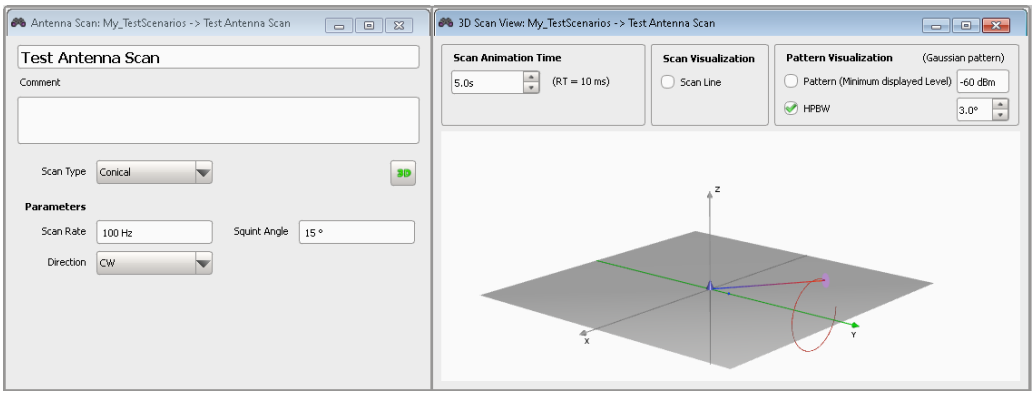
"Unidirectional" The antenna uses an unidirectional scan mode, i.e. the scan is not a continuous but the antenna turns in one direction only (counterclockwise, CCW) and "jumps" to the next bar within the selected "Flyback Time".

"Palmer Scan" If enabled, superimposes a conical scan on the circular scan, see figure 10-3.

Remote command:
`SCAN:RASTer:WIDTh` on page 296
`SCAN:RASTer:RATE` on page 293
`SCAN:RASTer:BARWidth` on page 295
`SCAN:RASTer:BARS` on page 294
`SCAN:RASTer:RETRace` on page 295
`SCAN:RASTer:DIRection` on page 296
`SCAN:RASTer:UNIDirection` on page 295
`SCAN:RASTer:FLYBack` on page 295
`SCAN:RASTer:PALMer` on page 298
`SCAN:RASTer:PRATe` on page 298
`SCAN:RASTer:PSQuint` on page 298

Conical Scan Settings

A conical antenna scans a conus with defined "Squint Angle" (radius); the radar beam is rotated around the y-axis (i.e. boresight).



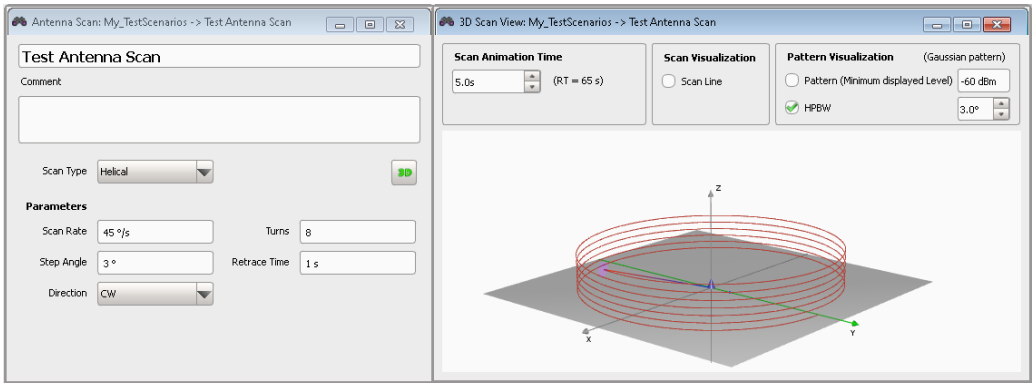
"Scan Rate, Hz"
Defines how fast the antenna is turning.

"Squint Angle, deg"
Sets the radius of the scanned circle.

"Direction" Sets the direction, clockwise ("CW") or counter clockwise ("CCW"), the antenna is turning in.

Remote command:
`SCAN:CONical:RATE` on page 293
`SCAN:CONical:ROTation` on page 292
`SCAN:CONical:SQUint` on page 294

Helical Scan Settings
A helical antenna scan turns in a circle (scans 360°) and performs several scans with different elevations.



"Scan Rate, deg/s"
Defines how fast the antenna is turning.

"Step Angle, deg"
Sets the step width the antenna azimuth is changing.

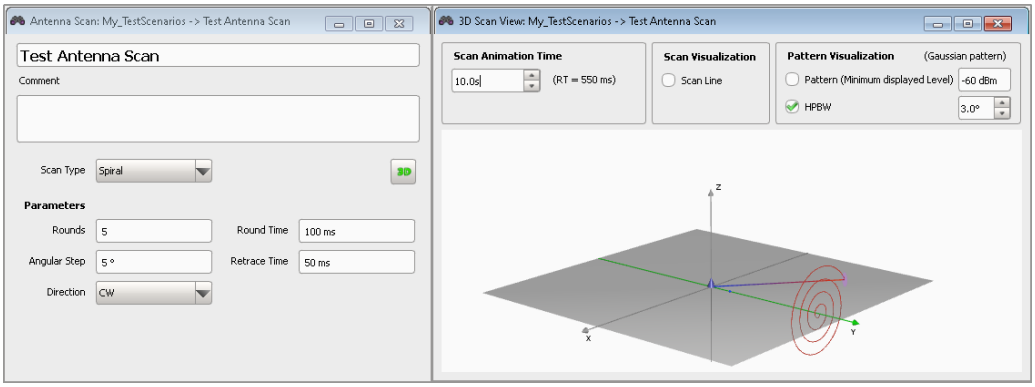
"Turns" Sets the number of 360° turns the antenna performs.

"Retrace Time" Defines how fast the antenna returns to its initial orientation after completing the last turn.

"Direction" Sets the direction, clockwise ("CW") or counter clockwise ("CCW"), the antenna is turning in.

Remote command:
`SCAN:HELical:RPM` on page 293
`SCAN:HELical:ELEVation:STEP` on page 293
`SCAN:HELical:TURNs` on page 293
`SCAN:HELical:RETRace` on page 295
`SCAN:HELical:ROTation` on page 292

Spiral Scan Settings
A spiral antenna scan turns in a circle (scans 360°) and performs several scans with different radiuses.



"Rounds" Sets the number of rounds the antenna performs.

"Angular Step" Sets the distance in degrees with that the scan radius is increased.

"Round Time" Defines how fast the antenna is turning.

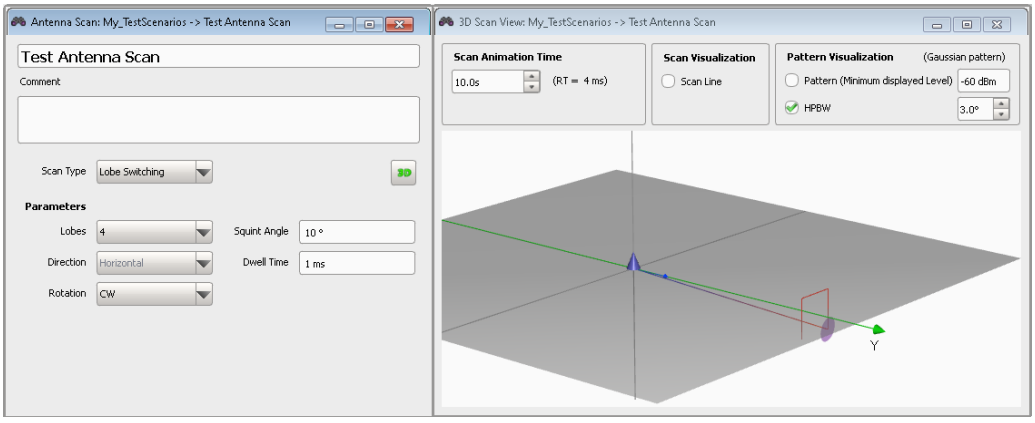
"Retrace Time" Define how fast the antenna returns to its initial orientation after completing the last turn.

"Direction" Sets the direction, clockwise ("CW") or counter clockwise ("CCW"), the antenna is turning in.

Remote command:
`SCAN:SPIRal:ROUNds` on page 296
`SCAN:SPIRal:STEP` on page 297
`SCAN:SPIRal:RTIMe` on page 296
`SCAN:SPIRal:RETRace` on page 295
`SCAN:SPIRal:ROTation` on page 292

Lobe Switching Scan Settings

A lobe switching antenna uses two or four slightly separated antenna elements.



"Lobes" Set the number of lobes.

"Direction" Sets the switching direction, i.e. horizontal or vertical.

"Squint Angle" Sets the angle that the beam axis is offset from the tracking axis (i.e. the y-axis) of the antenna.

"Dwell Time" Sets the antenna motion, i.e. how fast the switches between the lobes.

"Direction" Sets the direction, clockwise ("CW") or counter clockwise ("CCW"), the antenna is turning in.

Remote command:
SCAN:LSW:LOBes on page 294
SCAN:LSW:DIRection on page 294
SCAN:LSW:SQUint on page 294
SCAN:LSW:DWELl on page 294
SCAN:LSW:ROTation on page 292

10.4 How to Create a Library with Antenna Patterns and Scans

See:

- "To create an antenna pattern" on page 171
- "To create an antenna scan" on page 174
- "To create a custom antenna pattern based upon the required antenna characteristics" on page 175
- "To import a user-defined antenna pattern from file" on page 176

To create an antenna pattern

1. Select "Repository Tree > Antenna Pattern > New".
2. Enter a name and a comment.
3. Select "Antenna Model", e.g. "Planar Phased Array".



4. Adjust further settings, like "Rotation", "Frequency" or "Aperture Distribution".
5. Select "2D" to display the antenna pattern in a graph in polar or with Cartesian coordinates.

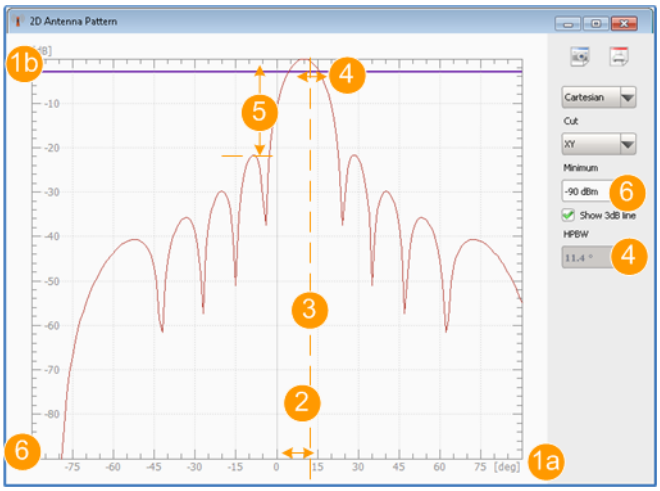


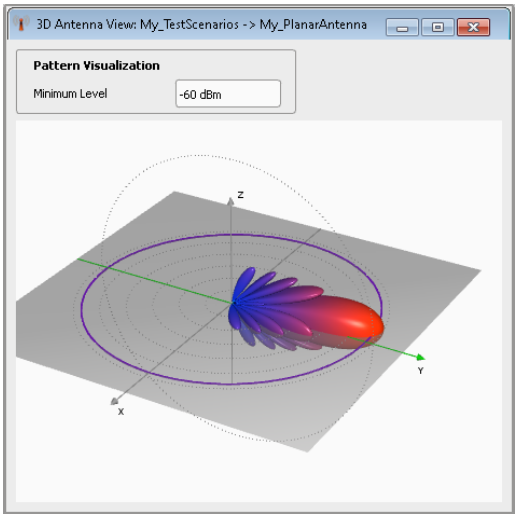
Fig. 10-4: 2D antenna pattern: understanding the displayed information

1a, 1b = Antenna pattern diagram, as a function of angle (deg) and antenna gain (dB)
2 = "Z-Rotation = 10°"
3 = Beam axis
4 = HPBW, i.e. the beam width at half power (3 dB)
5 = Side lobe level
6 = Antenna gain minimum (-90 dB)

6. Change the number of used elements, e.g. "Elements X = 20" and compare the 2D diagram.

More antenna elements result in a smaller beamwidth and side lobes with lower level.

7. Select "Antenna Pattern > 3D" to display the 3D antenna pattern.



The 3D view is interactive. Use the mouse cursor to turn it around any of the axis and the mouse wheel to zoom in the view.

8. Select "Antenna Pattern > Simulate Back Lobe > On" and "Attenuation = 0 dB".
9. In the 2D view, select "Polar" coordinates.

   The displays confirms the antenna pattern with a back lobe, where the back lobe is simulated as the mirrored pattern of the main lobe ("Type > Mirror").
   See figure 10-5.
10. Select "Antenna Pattern > Back Lobe Attenuation = 40 dB".

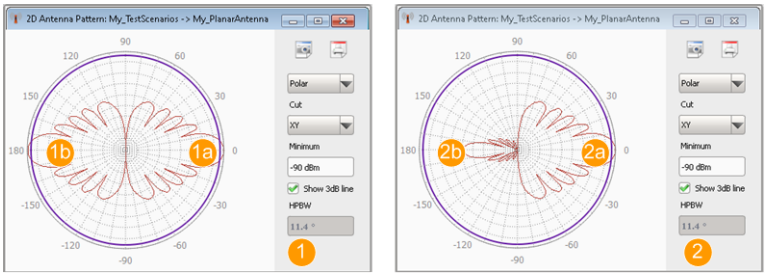


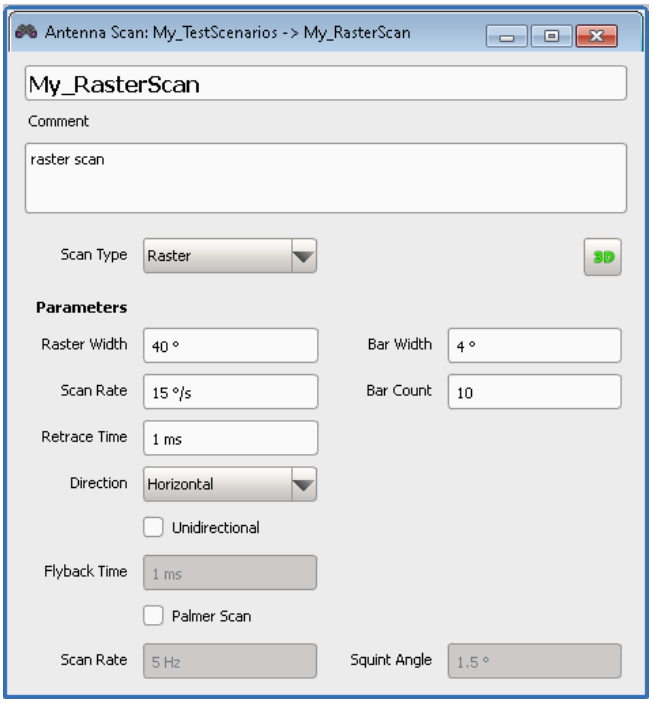
Fig. 10-5: Simulate Back Lobe: understanding the displayed information

1 = "Attenuation = 0 dB"
2 = "Attenuation = 40 dB"
1a, 2a = Main lobe
1b = Unattenuated back lobe, i.e. the mirrored pattern of the main lobe
2b = Attenuated back lobe

For description of the provided settings, see chapter 10.2, "Antenna Pattern Settings", on page 155.

To create an antenna scan

1. Select "Repository Tree > Antenna Scans > New".
2. Enter a name and a comment.
3. Select "Scan Type", e.g. "Raster Scan".



4. Adjust further settings, like "Raster Width" and "Bar Width".
5. Select "3D" to display the 3D antenna pattern.
6. Use the mouse cursor to turn it around any of the axis and the mouse wheel to zoom in the view.

   The antenna scan confirms the settings.

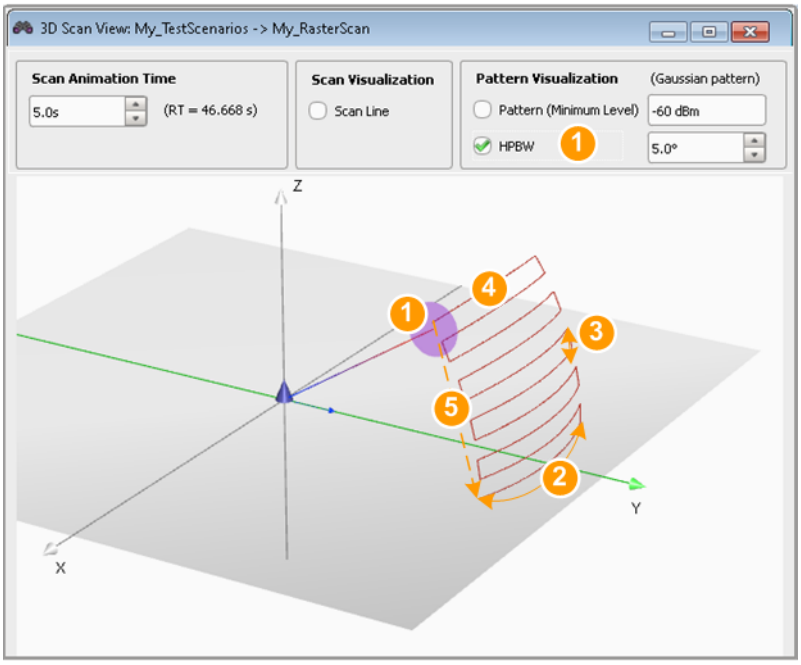


Fig. 10-6: Horizontal raster scan: understanding the displayed information

1 = Antenna pattern visualized as "HPBW = 5°"
2 = "Raster Width"
3 = "Bar Width = 4°"
4 = 10th scan as defined with "Bar Count"
5 = "Retrace Time"

For description of the provided settings, see chapter 10.3, "Antenna Scans Settings", on page 164.

To create a custom antenna pattern based upon the required antenna characteristics

1. Create an "Antenna Pattern".
   See "To create an antenna pattern" on page 171
2. Select "Antenna Model > Custom".
3. Adjust the settings, e.g. set the "HPBW XY = 10°", "HPBW YZ = 3°", etc.
4. Observe the antenna pattern on the 2D and 3D views.

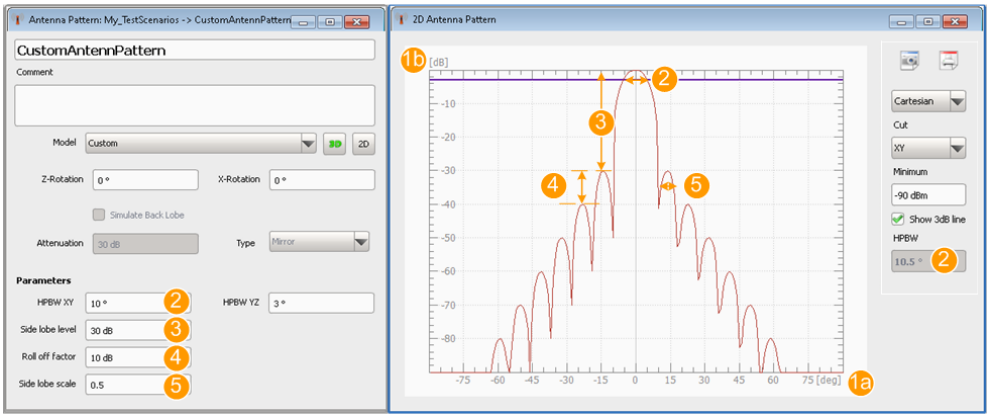


Fig. 10-7: Custom antenna pattern: understanding the displayed information

1a, 1b = Antenna pattern diagram, as a function of angle (deg) and antenna gain (dB)
2 = "HPBW XY = 10°", i.e. the main beam width at half power (3 dB)
3 = Side lobe level, i.e. the attenuation of the first pair of side lobes
4 = Roll off factor or step size, used to calculate the attenuation of the subsequent side lobes
5 = Scale factor to calculate the HPBW of the side lobes, i.e. how much smaller or wider the side lobes are compared to the main lobe

To import a user-defined antenna pattern from file

1. Create an "Antenna Pattern".
   See "To create an antenna pattern" on page 171
2. Select "Antenna Model > Import from File".
3. Select "Load", navigate to the folder with your custom antenna patterns and load a `*.ffe`, a `*.tsv`, a `*.ffd`, a `*.csv` or a `*.ant_pat` file.
   See chapter A.1.2, "Antenna Pattern File Formats", on page 393.
4. Open the 2D view to visualize the antenna pattern.

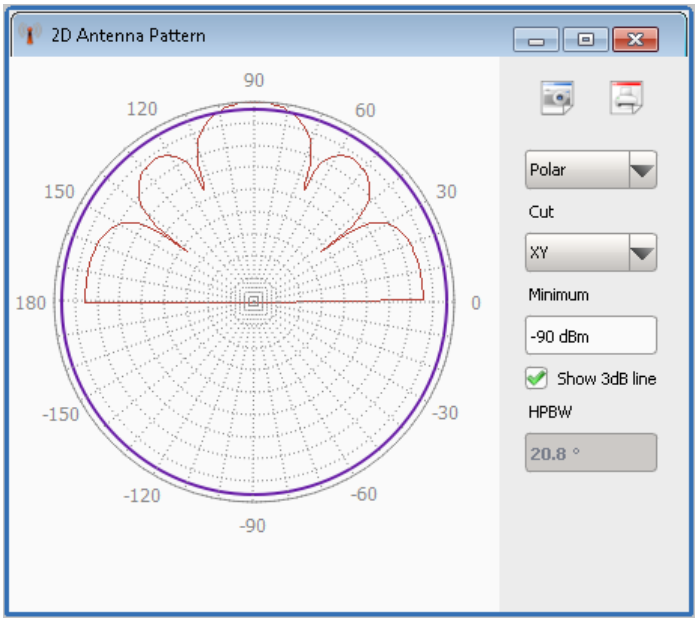
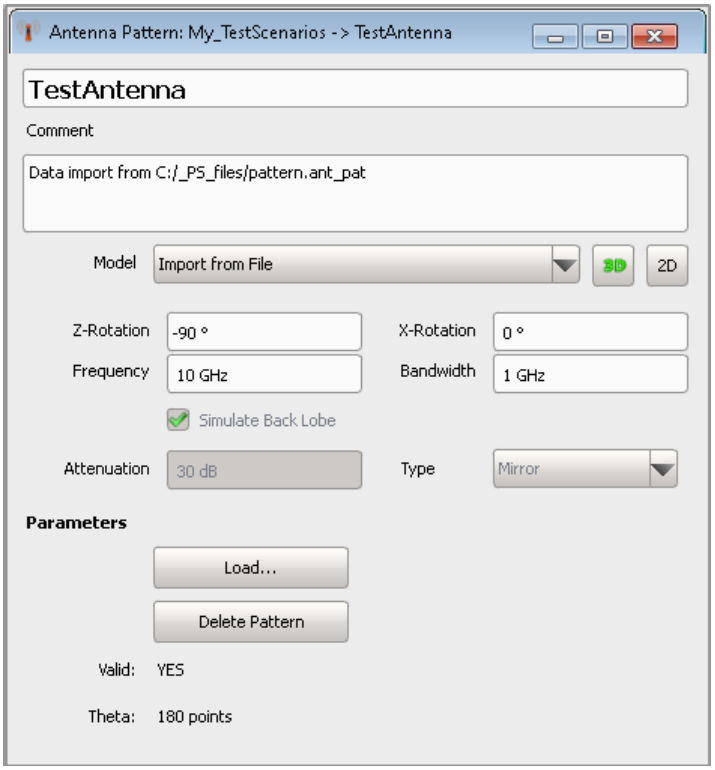


Fig. 10-8: 2D view: Example of a user-defined antenna pattern

In this example, the antenna pattern is directed to 90°.

5. Select "Antenna Pattern > Z-Rotation = -90°".



6. On the 2D and 3D views, observe the effect of this parameter.

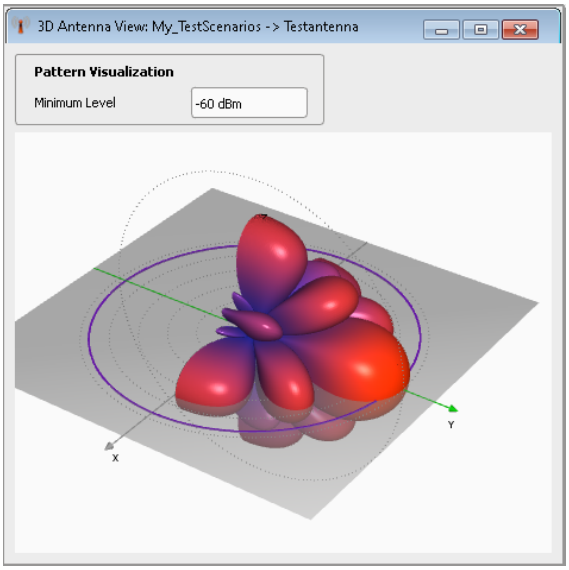


Fig. 10-9: 3D view: Example of a user-defined antenna pattern

The antenna pattern is directed on the Y axis.

11 Emulating Emitters

In the context of this software, a simple emitter describes a pulse sequence whereas a complex emitter emulates a radar system. The emitters are characterized with parameters like operating mode, beam, waveform, antenna pattern and antenna scan.

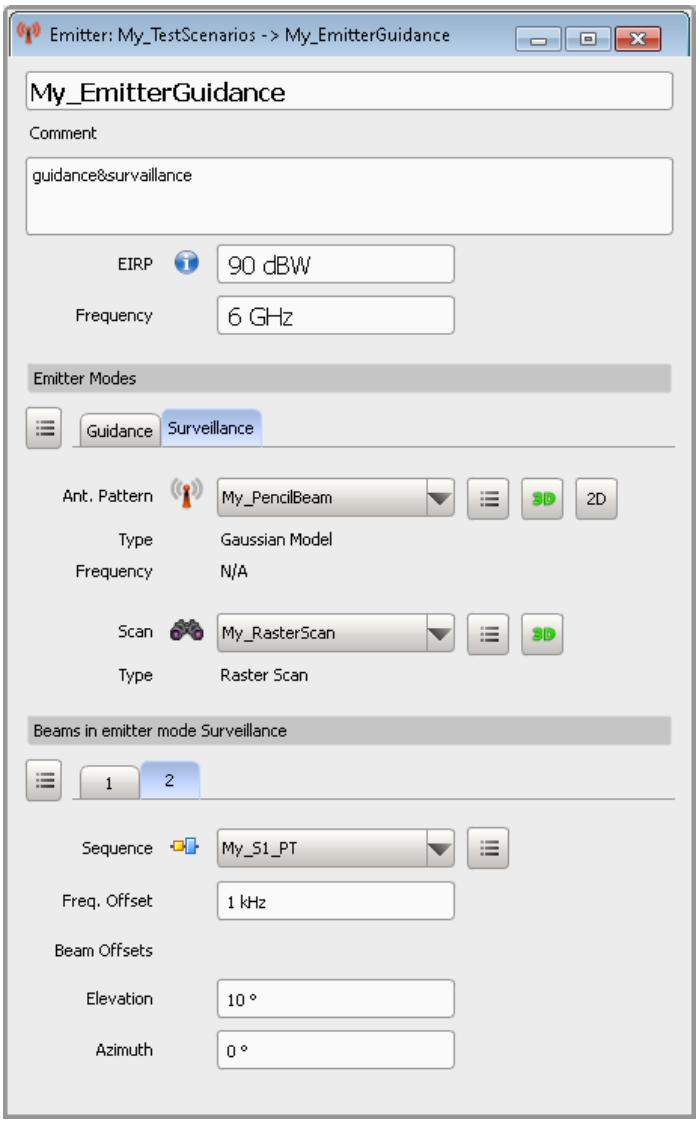
The available emitter settings depend on the complexity of the simulated system:

- Simple emitters are described only by a sequence of pulses. Antenna beam patterns and antenna scans are not required
- Complex emitters are described by the combination of a operating mode, antenna pattern, antenna scan and a sequence, as well as elevation and azimuth.

11.1 Emitter Settings

To access these settings:

► Select "Repository Tree > Emitter > New".



An emitter is described as a combination of a operating mode, an antenna pattern, an antenna scan and a sequence:

Emitter Name 181
Comment 181
EIRP 181
Frequency 181
Emitter Modes 181
Emitter Beams 182

Emitter Name
Enter the emitter's name.

Remote command:
EMITter:NAME on page 277
EMITter:CREate on page 276
EMITter:SELect on page 277
EMITter:CATalog? on page 276
EMITter:REMove on page 278

Comment
Enter a short description.

Remote command:
EMITter:COMMent on page 278

EIRP
Sets the equivalent isotopically radiated power (EIRP) of the emitter.

This parameter is used, if the emitter is configured in a "Scenario Type > Localized Emitter".

Example:

For emitter with:
- Transmit power = 1 MW
- Antenna Gain = 20 dBi

Set "EIRP = 120 dBW".

Remote command:
EMITter:EIRP on page 304

Frequency
Sets the emitter's operating frequency.

If the selected antenna is calculated for a particular frequency, set the emitter's "Frequency" to a value within the suggested frequency range, see "Emitter Modes" on page 181.

Remote command:
EMITter:FREQuency on page 305

Emitter Modes
Defines one or more modes the emitter works in.

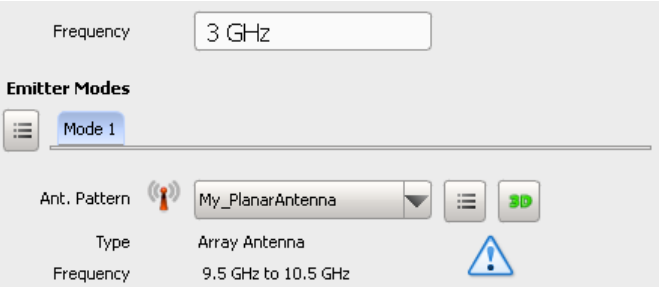
Use the standard functions in the context menu to create or rename modes.

Each emitter mode is defined by the combination of an antenna pattern, antenna scan and a beam with assigned pulse sequence.

Use the 3D view to visualize the selected antenna scan and pattern, see:
- "2D and 3D diagrams" on page 158
- "3D Scan View" on page 165

"Antenna Pattern"

Selects an existing antenna pattern or accesses a dialog for creating a new one.
See "To create an antenna pattern" on page 171.
The antenna pattern and type are displayed.
A warning symbol indicates that the selected emitter's Frequency is not sufficient for the current antenna.



Set the emitter's Frequency to a value within the suggested frequency range.

"Antenna Scan"

Selects an existing antenna scan or accesses a dialog for creating a new one.
See "To create an antenna scan" on page 174

Remote command:
`EMITter:MODE:NAME` on page 277
`EMITter:MODE:ADD` on page 279
`EMITter:MODE:COUNt?` on page 280
`EMITter:MODE:SELect` on page 280
`EMITter:MODE:CLEar` on page 282
`EMITter:MODE:DELete` on page 281
`EMITter:MODE:ANTenna` on page 305
`EMITter:MODE:ANTenna:CLEar` on page 281
`EMITter:MODE:SCAN` on page 306
`EMITter:MODE:SCAN:CLEar` on page 282

Emitter Beams

In each operating mode, the emitter may use one or more beams (e.g. different pulse sequences), while using the same antenna and scan type.

Use the standard functions in the context menu to create or rename beams.

Assign a pulse sequence to each beam.

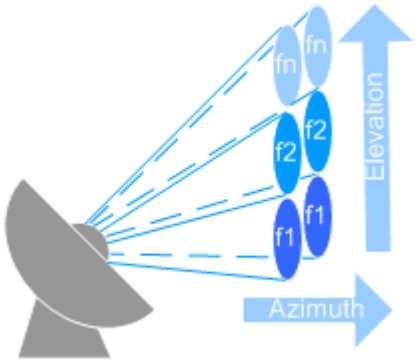


Fig. 11-1: Stacked beams (simplified representation)

"Activate" Activates a beam.

"Sequence" Selects an existing sequence or accesses a dialog for creating a new one.
See chapter 8.3, "How to Create Sequences and Use the Control Elements", on page 117

"Frequency Offset"
Offsets the beam frequency by the selected value, see figure 11-1.

"Beam Position Offset"
Offsets the beam in terms of an "Elevation" and an "Azimuth", see figure 11-1.

To *offset the complete emitter's antenna*, use one of the "Elevation" parameters, depending on your current scenario:

- "Single Emitter" scenario: Elevation
  (see "To configure an emitter in a scenario with static receiver" on page 188)
- "Emitters (Collection)" scenario: Elevation
  (see "To configure complex scenarios with several emitters" on page 189)
- Localized Emitters scenario: Elevation
  (see "To enable static emitters and place them on the 2D map" on page 211)

Remote command:
`EMITter:MODE:BEAM:ADD` on page 279
`EMITter:MODE:BEAM:COUNt?` on page 280
`EMITter:MODE:BEAM:SELect` on page 280
`EMITter:MODE:BEAM:STATe` on page 306
`EMITter:MODE:BEAM:SEQuence` on page 305
`EMITter:MODE:BEAM:OFFSet:AZIMuth` on page 305
`EMITter:MODE:BEAM:OFFSet:ELEVation` on page 305
`EMITter:MODE:BEAM:OFFSet:FREQuency` on page 305
`EMITter:MODE:BEAM:CLEar` on page 281
`EMITter:MODE:BEAM:DELete` on page 281

11.2 3D Emitter Preview Settings

This dialog display:

- a 3D view of the emitter antenna with its pattern and scan
- a live plot of the normalized signal power level at the receiver

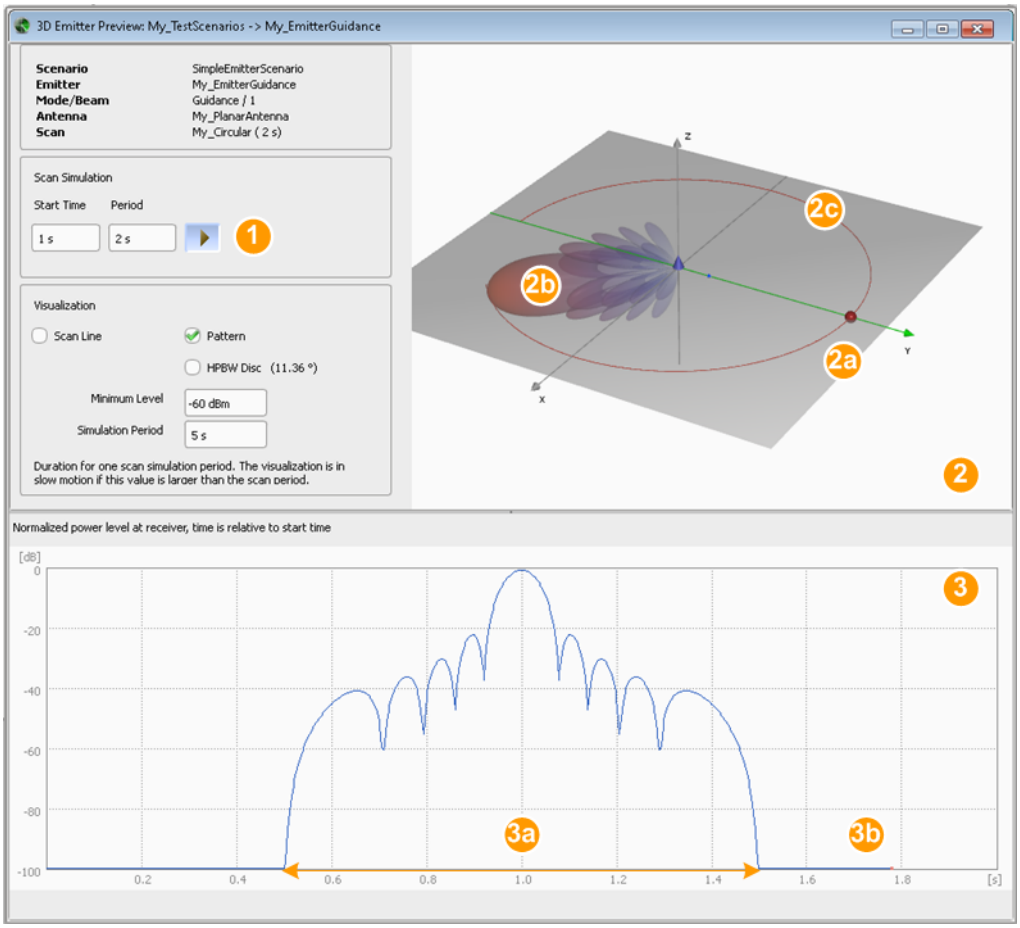


Fig. 11-2: 3D Emitter Preview: Understanding the displayed information

1 = Selection of preview time span (selected is one antenna turn)
2 = Interactive 3D view
2a = Receiver with an isotropic antenna pattern, no antenna scan and at a fixed location
2b = Antenna pattern diagram of the emitter; antenna pattern without back lobes
2c = Circular antenna scan
3 = Live plot of the normalized signal power level at the receiver
3a = Antenna pattern (main and side lobes), compare with the pattern on figure 10-4
3b = Antenna pattern without back lobes

Use the following setting to adjust the display:

Scan Simulation..... 184
Visualization..... 185

Scan Simulation
Sets the time span visualized on the live plot as a start time and simulation period.

The "Period", i.e. the total duration the antenna scan simulation, determines the max. value on the x-axis. The visualization is relative to the selected "Start Time".

Use the "Pause" icon to stop the visualization.

Visualization
Defines the way the scan is represented:

"Scan Line" The scan is visualized by a line; the antenna pattern is disregarded.

"Pattern" The visualization corresponds to the current selected antenna pattern.

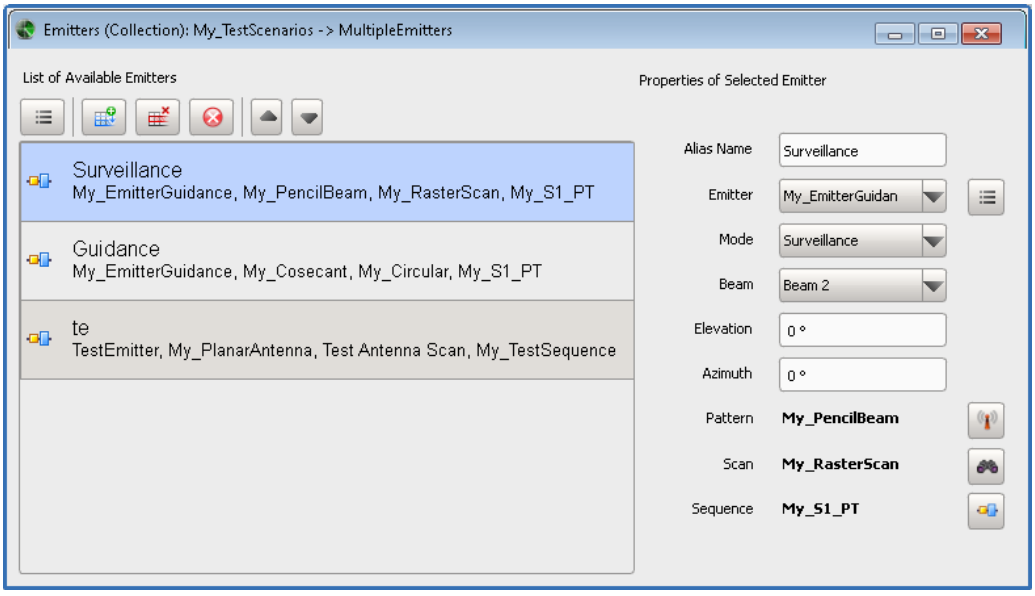
"HPBW" The antenna is represented by its HPBW.
See also "To visualize the signal received by a static receiver" on page 188.

"Minimum displayed level"
Defines the minimum displayed side lobes level.

"Simulation Period"
Sets the time it takes the animation to complete a scan.

11.3 Lists with Multiple Emitters

The "Emitter (Collection)" scenario comprises several emitters. The available elements are described in a list form.



You can configure the following "Emitter Properties":

List of Available Emitters.....185
Properties of the selected Emitter.....186

List of Available Emitters
Displays a list of emitters.

Use the standard functions in the context menu to add, reorder or remove items.

See also "To configure complex scenarios with several emitters" on page 189.

Remote command:
SCENario:CEMit:ADD on page 279
SCENario:CEMit:SELect on page 280
SCENario:CEMit:DELete on page 281

Properties of the selected Emitter
Each emitter is described with the following parameters:

"Alias Name" Enters an alias name.

"Emitter Name"
Selects an emitter/interferer.

"Mode" Selects the mode the emitter is working in.

"Beam" Sets the number of the currently used beam.

"Elevation, Azimuth"
Offsets the antenna in terms of an "Elevation" and an "Azimuth".

"Antenna Pattern, Scan, Sequence"
Displays the current selected antenna pattern, antenna scan and sequence.
To change any of them, select the corresponding icon.

Remote command:
SCENario:CEMit:ALIas on page 363
SCENario:CEMit:EMITter on page 364
SCENario:CEMit:EMITter:MODE on page 365
SCENario:CEMit:EMITter:MODE:BEAM on page 365
SCENario:CEMit:DIRection:ELEVation on page 364
SCENario:CEMit:DIRection:BEARing on page 364

11.4 How to Create and Configure Emitters

See:

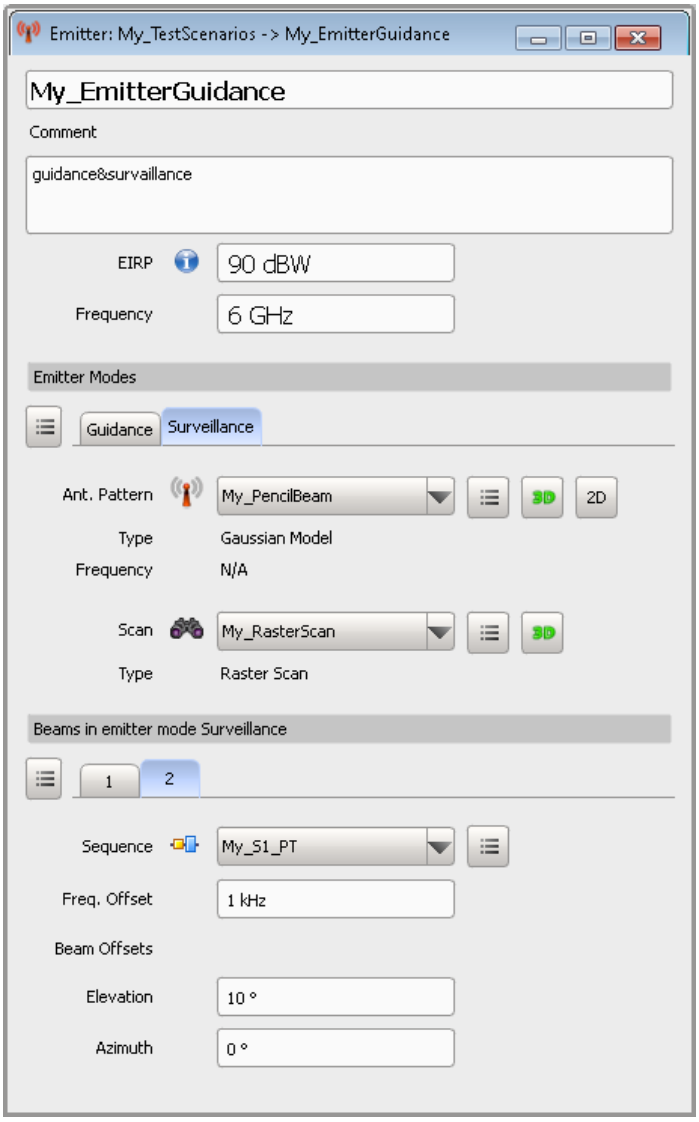
- "To create and configure a new emitter" on page 186
- "To configure an emitter in a scenario with static receiver" on page 188
- "To visualize the signal received by a static receiver" on page 188
- "To configure complex scenarios with several emitters" on page 189
- "To enable static emitters and place them on the 2D map" on page 211

To create and configure a new emitter

You can clone or copy an existing emitter or create an new one:

1. Select "Repository Tree > Emitter > New".
2. Enter a name and a comment.

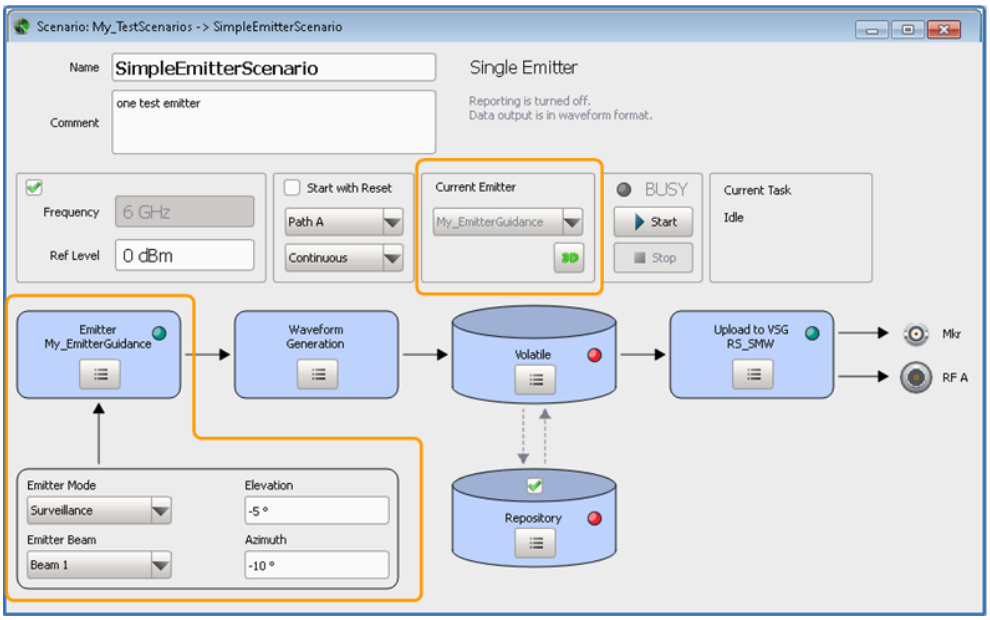
3. Select "Mode 1", open the context menu and select "Rename".
4. Enter a name for the emitter mode, e.g. "Guidance".
5. Select "Ant. Pattern" and select an antenna pattern from the antenna library or create a new one.
   See "To create an antenna pattern" on page 171
6. Select "Scan Type" and select an antenna scan from the antenna library or create a new one.
   See "To create an antenna scan" on page 174
7. To configure the "Beam":
   a) Select "Activate > On"
   b) Select "Sequence" and select a sequence from the library.
8. If required, enable a second beam.



For description of the provided settings, see chapter 11.1, "Emitter Settings", on page 180.

To configure an emitter in a scenario with static receiver

1. Create a new "Single Emitter" scenario or open an existing one, e.g. "SimpleEmitterScenario".
   See for example "To create a single pulse train scenario" on page 64.



2. In the block diagram, select "Emitters > Data Source Menu" and select an emitter from the list of available emitters or create a new one.
   See also "To create and configure a new emitter" on page 186.
3. Configure the emitter settings, like for example select the current "Emitter Mode" and "Emitter Beam".

   For description of the provided settings, see chapter 11.3, "Lists with Multiple Emitters", on page 185.

To visualize the signal received by a static receiver

1. Open a scenario with single emitter, e.g. "SimpleEmitterScenario".
   See also "To configure an emitter in a scenario with static receiver" on page 188.
2. In the block diagram, select "Current Emitter > 3D".

   The dialog displays the signal as "seen" by a static receiver. Per default, the receiver is located in a way, that the emitter scan is oriented with 0 deg "Elevation" and "Azimuth" at it.
3. To "move" the receiver on the x-axis and on the z-axis, enable "Scenario > Elevation" and "Scenario > Azimuth". The position on the y-axis (the distance to the emitter) is fixed.

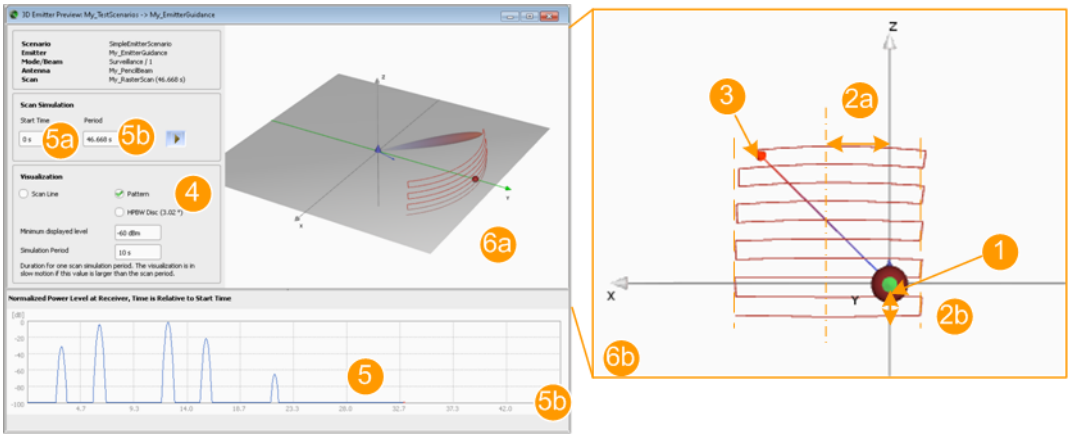


Fig. 11-3: Signal at the receiver (XZ view): understanding the displayed information

1 = Receiver's position, determined by the selected "Azimuth" and "Elevation"; the position on the y-axis (the distance to the emitter) is fixed
2a = "Azimuth = -10 deg" turns the antenna horizontally, i.e. the receiver is not at the central scan axis but rotated in clockwise (CW) direction on the x-axis
2b = "Elevation = -5 deg" turns the antenna vertically, i.e. "moves" the receiver on the z-axis
3 = Current position of the scanning beam
4 = Scan and pattern visualization; 6a uses "Scan Line = Pattern"
5 = Amplitude changes in the received signal over time
5a, 5b = Start time and simulation period (max value on the x-axis)
6a, 6b = 6b is the XZ representation of the 6a 3D view; 6b uses "Scan Line = On"

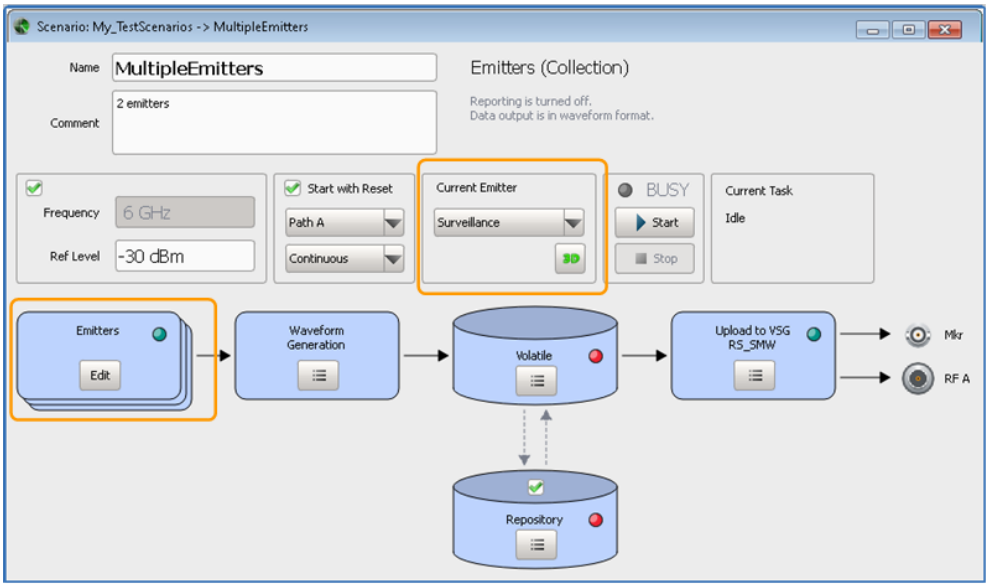
4. Use the mouse to change the orientation of the interactive 3D diagram and to zoom on it.
   For description of the provided settings, see chapter 11.2, "3D Emitter Preview Settings", on page 184.

To configure complex scenarios with several emitters

If you test situation requires to test the receiver's ability to correctly detect the signal from different static emitters, you can use "Emitter (Collection)" scenario. In this dedicated scenario, you can configure several emitters and switch between them sequentially. You can also configure the receiver and change its position in the scanning beam of the emitters.

The following example create a complex scenarios with several emitters and a receiver to explain the principle.

1. Create a new "Emitter (Collection)" scenario or open an existing one, e.g. "MultipleEmitters".
   See for example "To create a single pulse train scenario" on page 64.



2. In the block diagram, select "Emitters > Edit".

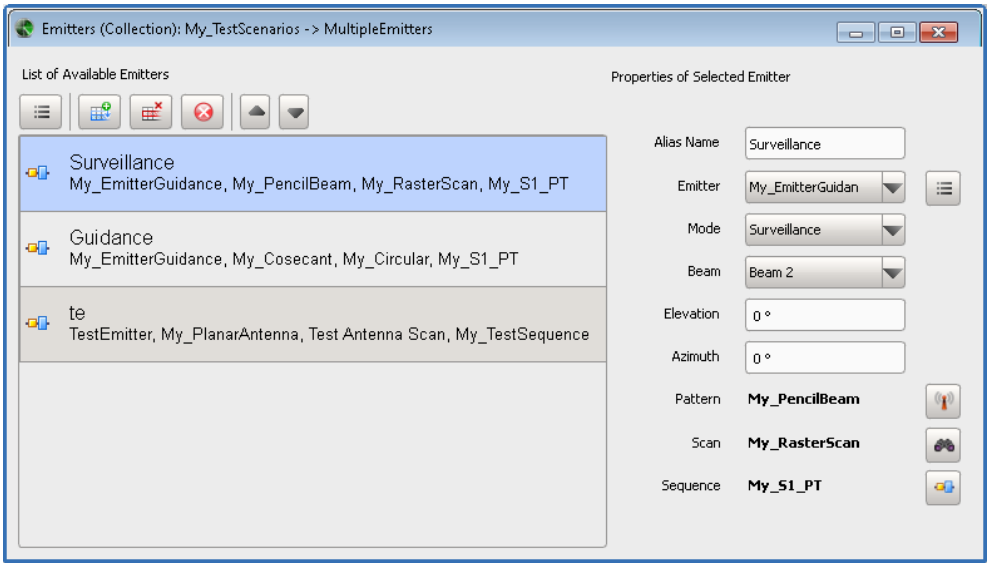
   The "Multiple Emitter" dialog opens.

3. Use the standard "New" function and insert the first emitter in the list.

   You can select from the list of available emitters or create a new one.
   See also "To create and configure a new emitter" on page 186.

4. Configure the emitter settings, like for example enter an alias name.

5. In the list of emitters, select a "Emitter" and use the standard "New", "Append/ Remove" and "Up/Down" functions to add emitters to the list and reorder them.



6. In the "Scenario" dialog, select the emitter that is currently transmitting, e.g. "Surveillance".

7. Use the 3D view to visualize the signal received by a static receiver.

See figure 11-3.

For description of the provided settings, see:

- chapter 11.3, "Lists with Multiple Emitters", on page 185
- chapter 11.2, "3D Emitter Preview Settings", on page 184

12 Working with Waveforms and Generating Interfering Signals

Additionally to creating pulse signals you can also load waveform files and create waveform scenarios with them or even use them as interferers.

An waveform can be calculated by the R&S Pulse Sequencer or created with an external software. As long as the data format is in one of the supported import formats, you can import any waveform into the waveform library.

Supported waveform types

The fallowing types are supported:

- Internally created:
  CW waveforms or AWGN waveforms with defined frequency and bandwidth.
- Externally created:
  – Waveforms in the Rohde&Schwarz proprietary file format (`*.wv`); such files are generated with the signal generation software R&S WinIQSIM2 or with the real-time options of the Rohde&Schwarz signal generators.
  – Files with custom I/Q data created with other Rohde&Schwarz instruments or third party software, like MATLAB (`*.mat`), R&S® PR100 (`*.riq`), Rohde&Schwarz signal and spectrum analyzers (`*.iq.tar`) or the Rohde&Schwarz proprietary modified audio file format (`*.wav`).

  See also table 1-1.

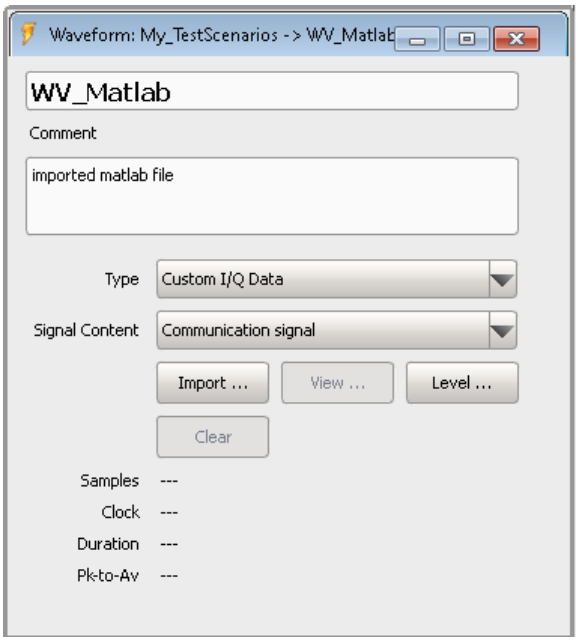
The sequences containing wavefroms are also described in table form where the rows are processed sequentially. Each row is repeated once or as defined with the parameter "Repetition Count". Complex waveform sequences may contain waveforms and loops. Other control elements are not supported.

See chapter 8.2.1, "Sequence Description Settings", on page 109.

12.1 Waveform Settings

To access these settings:

► Select "Repository Tree > Waveform > New".



The following settings are available:

Waveform Name..... 193
Comment..... 194
Type..... 194
Signal Content..... 194
Bandwidth..... 194
Import..... 194
Clear..... 194
View, Level..... 194

Waveform Name
Enters a name.

Remote command:
`WAVeform:NAME` on page 278
`WAVeform:CREate` on page 277
`WAVeform:SELect` on page 277
`WAVeform:CATalog?` on page 276
`WAVeform:REMove` on page 279

Comment
Adds a description.

Remote command:
WAVeform:COMMent on page 278

Type
Sets the type of the waveform to be created or imported, see "Supported waveform types" on page 192.

Remote command:
WAVeform:TYPE on page 384

Signal Content
Defines the waveform signal type, so that so that during the resampling process, different filters are applied on pulsed signals and on the waveforms containing digital standard signals.

Waveforms containing digital standard signals are usually intended for continuous playback. The end of the waveform fits the beginning to avoid spectral regrowth caused by the wrap-around.

For pulsed signals, however, wrap-around and overshoot effects due to steep filters are undesired. Thus, low order filters are applied to prevent from overshoots.

Remote command:
WAVeform:SIGCont on page 385

Bandwidth
Sets the bandwidth of the generated AWGN waveform.

Remote command:
WAVeform:NOISe:BWIDth on page 384

Import
Accesses a file import wizard.

Remote command:
WAVeform:WAVeform:LOAD on page 385

Clear
Removes the imported waveform or file with I/Q data.

Remote command:
WAVeform:WAVeform:CLEar on page 385
WAVeform:IQ:CLEar on page 385

View, Level
Access dialogs which visualize the waveform and displays more useful information on it, see:
- chapter 18.1, "Waveform and Data View Settings", on page 240
- chapter 18.2, "Waveform Reference Level Settings", on page 243

Remote command:
WAVeform:LEVel:REFerence on page 385

12.2 How to Create a Waveform Scenario and Work with Waveforms

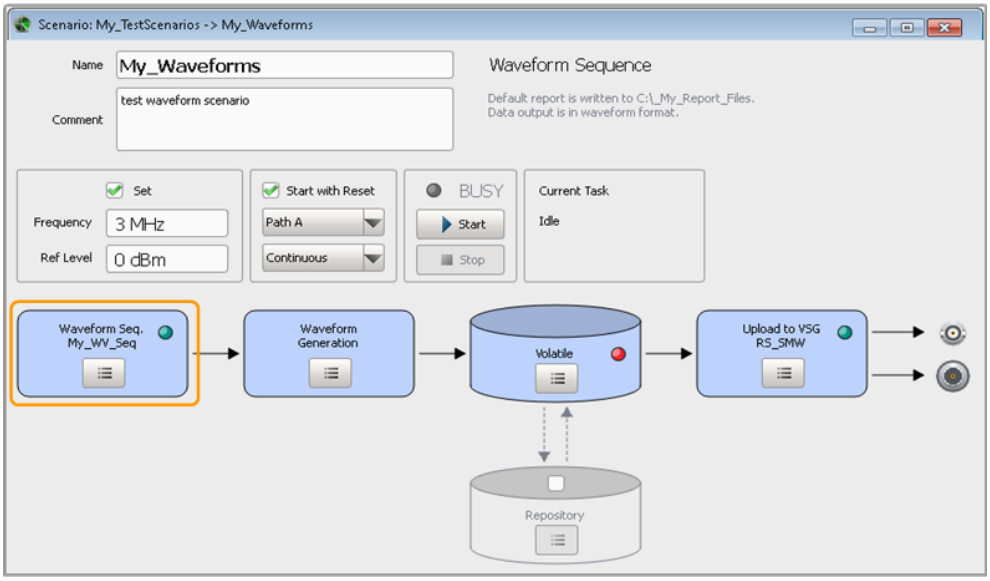
See:

- "To create a waveform scenario" on page 195
- "To create a new sequence of waveforms" on page 195
- For how to build an waveform library, see:
  - "To create a new waveform" on page 196
  - "To import a custom waveform" on page 197
  - "To retrieve more information on an imported waveform" on page 199
- "To simulate an interferer" on page 200

The "Waveform Sequence" scenario is a dedicated scenario type for processing wavefroms.

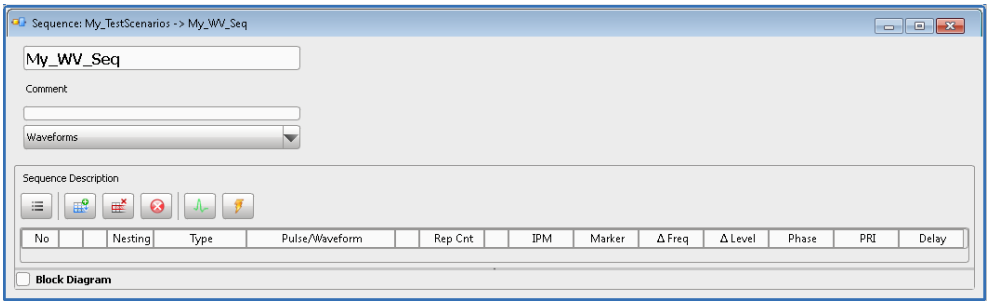
To create a waveform scenario

► Create a new scenario with "Scenario Type > Waveform Sequence".
See also "To create a single pulse train scenario" on page 64.



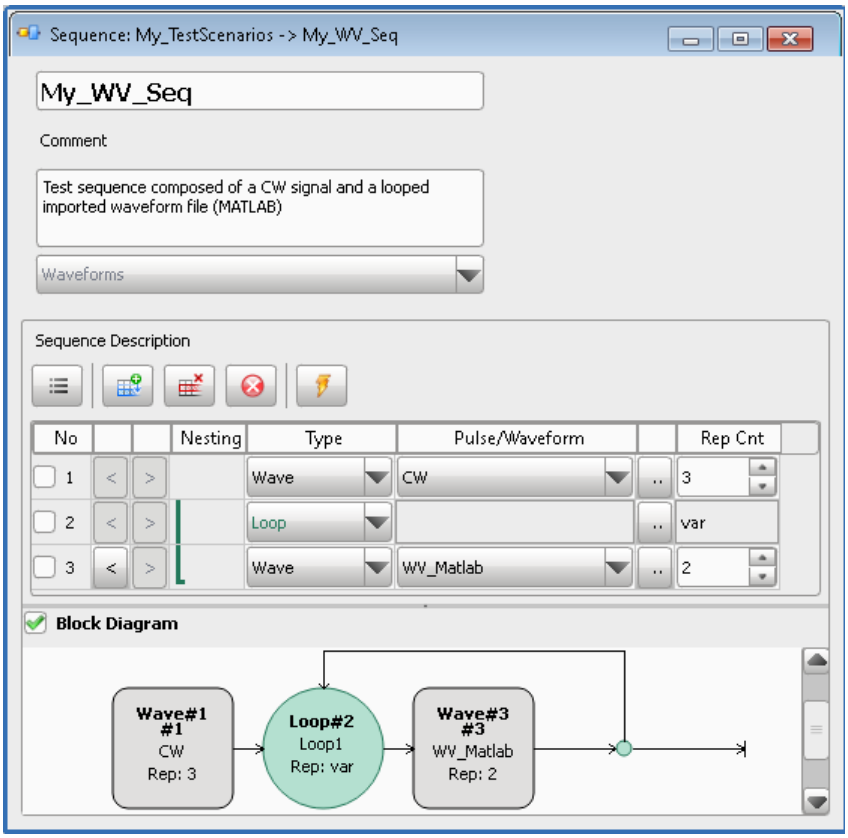
To create a new sequence of waveforms

1. Perform one of the following:
   a) In the "Waveform Scenario" dialog, select "Waveform Sequence > New".
   b) Select "Repository Tree > Sequence > New" and select "Sequence Type > Wavefroms".



2. To add waveforms to the sequence, perform one of the following:
   a) select the waveform icon to create a new waveform, see "To create a new waveform" on page 196.
   b) If there are waveforms in your repository, select the "Append New Item" icon and select "Type > Wave".

   A waveform sequence contains one ore more waveform. Complex sequences may also contain loops.



   See also "To define and apply a loop" on page 119.

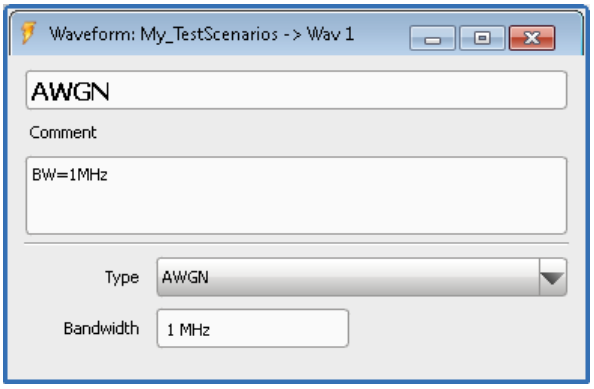
To create a new waveform

1. Select "Repository Tree > Waveform > New".

   A new waveform with default settings is created and added to the project tree.

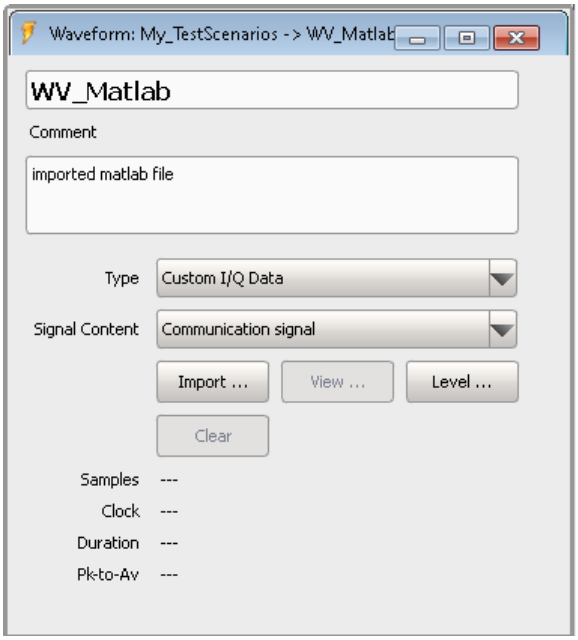
New waveforms are named `Wav_<n>` where `n` is a number starting at one.

2. Add information to describe and identify the waveform, like a name or a comment.
3. Select "Type > AWGN" and set the "Bandwidth".
4. Change the waveform name, e.g. "AWGN".



To import a custom waveform

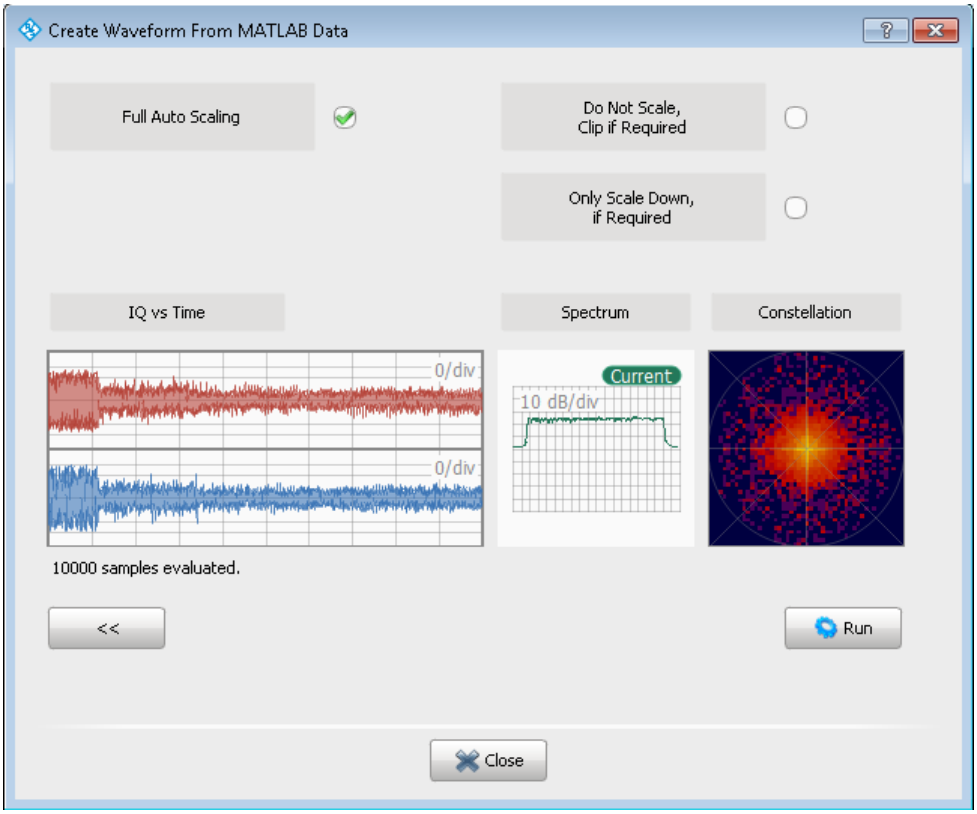
1. Select "Repository Tree > Waveform > New".
2. Enter a name and a comment.
3. Select "Type > Custom I/Q Data".



4. Select "Import", navigate and select the `*.mat` file and follow the instructions.

   A "Import Wizard" opens. The wizard differs depending on the selected file format but the differences are self-explanatory.

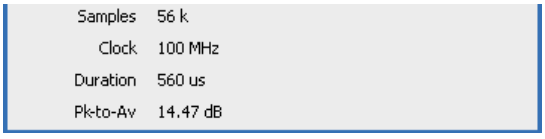
The software retrieves important parameters like used clock rate, detects whether the file with I/Q data contains one or more packets and per default scales automatically the waveform.
The waveform is calculated so that it perfectly fits the 16 bit dynamic range of the instrument's ARB. The waveform is not clipped.



The dialog shows also a preview of the waveform or, for large waveforms, of the beginning portion of the waveform.

5. Only if your particular application requires it, change the scaling level:
   a) Select "Do Not Scale, Clip if Required > On" to leave the signal amplitude unchanged.
      This could lead to low signal levels or clipping.
   b) Select "Only Scale Down, if Required > On"

After the import have been completed, the "Waveform" dialog displays information on the waveform.



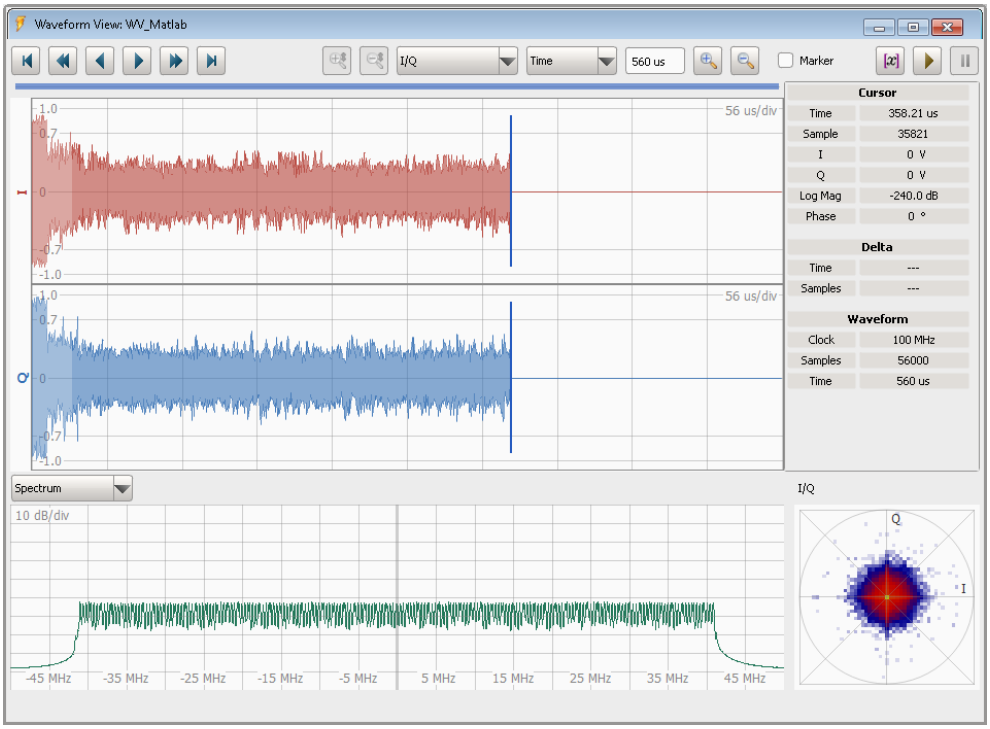
- **Samples**: Indicates the number of samples that the waveform contains. A warning symbol indicates that the waveform contains less than 512 samples. The waveform length might violate the minimum required ARB file size.
- **Clock**: Indicates the used clock rate.

- **Duration**: Indicates the waveform duration
- **Peak-to-Average**: Shows the estimated reference level of the evaluated waveform section.

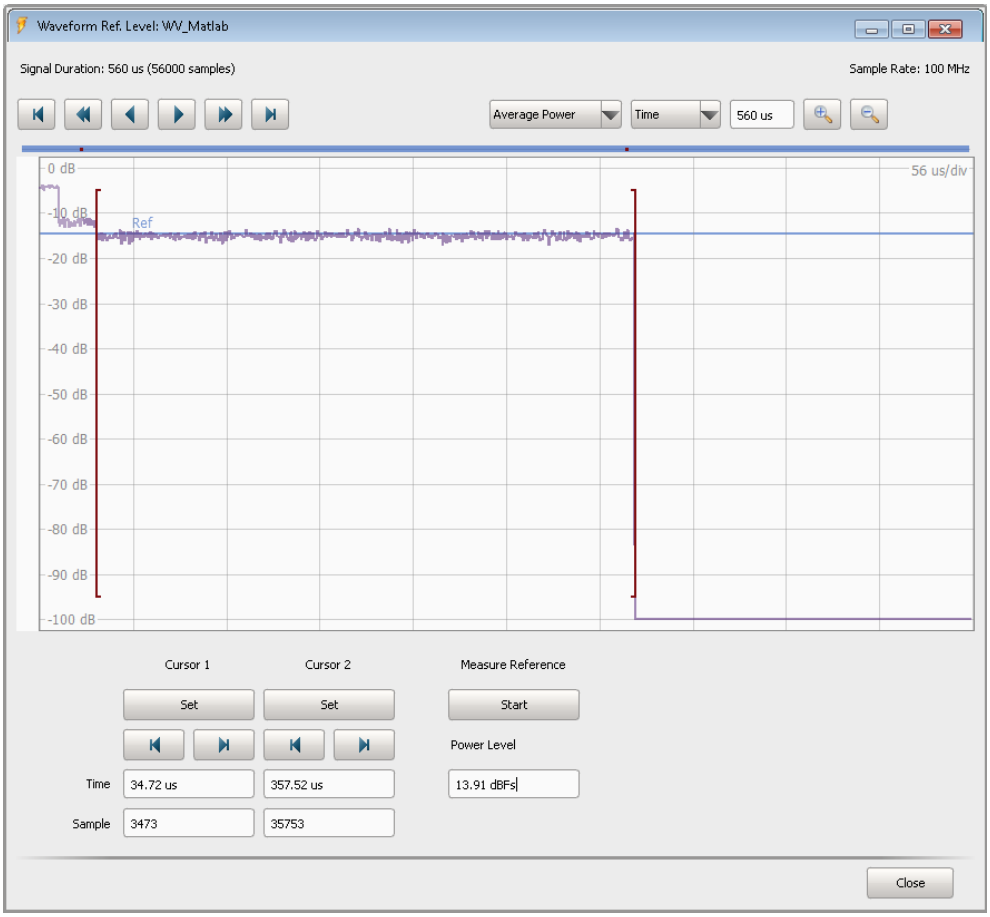
See also "To retrieve more information on an imported waveform" on page 199.

To retrieve more information on an imported waveform

1. Select "Repository Tree > Waveform", e.g. "WV_Matlab"
2. Select "View" to display for example the I/Q diagram or the spectrum of the signal.



3. Select "Level" to display information concerning the level, for example to measure the reference level.



For description of the provided settings, see:

- chapter 18, "Visualizing and Analyzing Waveforms", on page 240
- chapter 18.1, "Waveform and Data View Settings", on page 240
- chapter 18.2, "Waveform Reference Level Settings", on page 243

To simulate an interferer

Waveforms can be processed as sequence or, in "Localized Emitters" scenarios, used to simulated interferes. In these scenarios, interferers are treated like emitters, i.e. they are described with an waveform, an antenna pattern and antenna scan, static location with it X, Y and Z coordinates, and orientation defined with an "Azimuth" and an "Elevation".

► See "To add an existing waveform as an interferer to the scenario" on page 210 for how to simulate an interferer.

13 Creating Complex 2D Scenarios with Receivers and Interferers

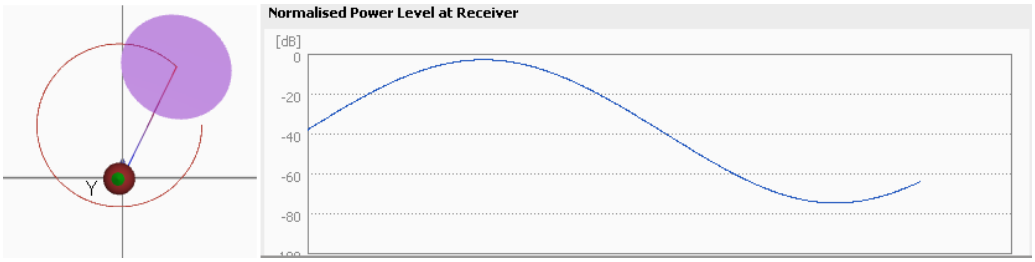
This section explains the interferer and receiver settings, and provides a brief information on radar systems.

Receiver

One of the typical receiver characteristic is the Minimum Detective Signal (MDS). The MDS is a measure of the receiver sensitivity and describes the minimum received signal amplitude S_{min} that the receiver is able to correctly detect. A typical receiver achieves a S_{min} within the range of -80 dBm and -110 dBm.

In the R&S Pulse Sequencer, the receiver is described by an antenna pattern, antenna scan, Gain, and a pointing direction defined as azimuth and elevation.

In a simple emitter-receiver scenario with static receiver, the R&S Pulse Sequencer calculates the received signal and displays the time variation of the received normalized power.



The signal received by a static receiver has the shape of a sine wave, where the amplitude is proportional to the distance between the target and the beam axis.

The principle is illustrated on figure 13-1.

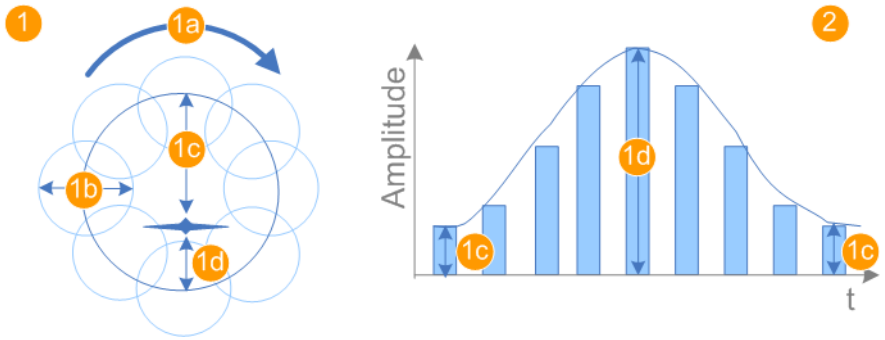


Fig. 13-1: Simplified representation of a conical scan and the received signal

1 = Conical scan (antenna beams foresight)
1a = Direction of the rotating scan
1b = Antenna beam represented as its HPBW
1c, 1d = Target to beam distance, determines the amplitude of the received signal
2 = Variation of the amplitude of the received signal

See:

- "To visualize the signal received by a static receiver" on page 188
- "To configure the receiver settings" on page 213

Interferers

Waveforms can be processed as sequence or, in "Localized Emitters" scenarios, they can be used to simulate interferers. In these scenarios, interferers are treated like emitters, i.e. they are described by a waveform, an antenna pattern and antenna scan, static location with it X, Y and Z coordinates, and orientation defined as "Azimuth" and "Elevation".

See:

- "To add an existing waveform as an interferer to the scenario" on page 210
- "To visualize the signal on a 3D scan" on page 213

13.1 Related Settings

- 2D Map Settings....202
- Receiver Settings....204
- 3D Scan Pair View Settings....205
- Lists with multiple emitters and interferers....206

13.1.1 2D Map Settings

To access this dialog:

► In a "Localized Emitters" scenario, select "Emitters > 2D".

The 2D diagram displays the configuration on a map with North/East coordinates.

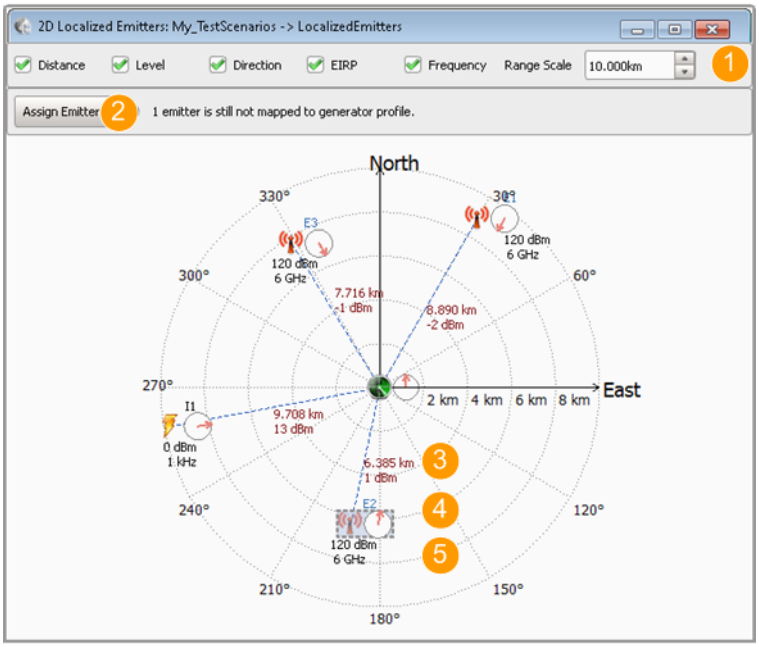


Fig. 13-2: 2D view: understanding the displayed information

1 = Display options; the "Range Scale" for example defines the radius of the displayed 2D map
2 = Access the "Assign Signal Source to Generator Profiles" dialog and indicates the mapping status
3 = Distance to the receiver and Strength of the received signal
4 = "Point to Receiver = On" (antenna is automatically turned in the direction of the receiver)
5 = Emitter (or interferer) indicated with its alias name, EIRP and transmission frequency

Distance, Level, Direction, EIRP, Frequency 203
Range Scale 203
Assign Emitters 203
Strength of the received signal 203
Emitter/Interferer position on the 2D map and distance to the receiver 204

Distance, Level, Direction, EIRP, Frequency
To change the displayed information, enable or disable the corresponding parameter.

Range Scale
Sets the radius of the 2D map.

Assign Emitters
Accesses the "Assign Signal Source to Generator Profiles" dialog (see chapter 19.2, "Signal to Generator Mapping Settings", on page 250).

Strength of the received signal
The displayed value is the reference value, used to calculate the power at the receiver antenna output. It is the virtual power level at the receiver output for two *perfectly aligned isotropic antennas* of emitter and receiver. Hence, this value is the maximum signal level that can be expected at the receiver antenna output.

In practice, the received signal will be weaker than the displayed signal strength. The following parameters influence the signal strength:

- Distance between the emitter and the receiver, i.e. the free space attenuation
- Alignment of the antennas and used antenna patterns
- Antenna gain

Emitter/Interferer position on the 2D map and distance to the receiver
The 2D map indicates the current distance, calculated from the current emitter/interferer position on the 2D map.

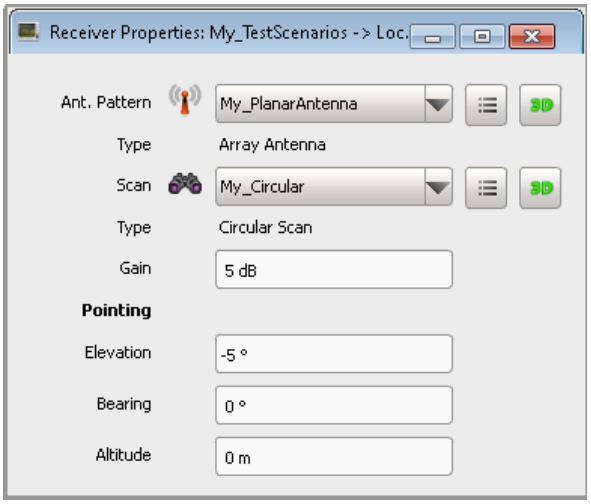
To change the position of the emitter/interferer, use one of the following possibilities:

- drag the emitter/interferer to the new position on the 2D map
- open its context menu, select "Properties > Location" and change the "East/North" coordinates, the "Altitude" and the "Distance".

13.1.2 Receiver Settings

To access this dialog:

1. In a "Localized Emitters" scenario, select "Emitters > 2D".
2. In the 2D map, open the context menu of the "Receiver" and select "Properties".



A receiver is describe with:

Antenna Pattern and Scan.....204
Gain.....205
Pointing Direction.....205

Antenna Pattern and Scan
Selects an antenna pattern and antenna scan type.

See also "To configure the receiver settings" on page 213.

Remote command:
SCENario:LOCalized:RECeiver:ANTenna on page 366
SCENario:LOCalized:RECeiver:SCAN on page 367

Gain
Sets the antenna Gain.

Remote command:
SCENario:LOCalized:RECeiver:GAIN on page 367

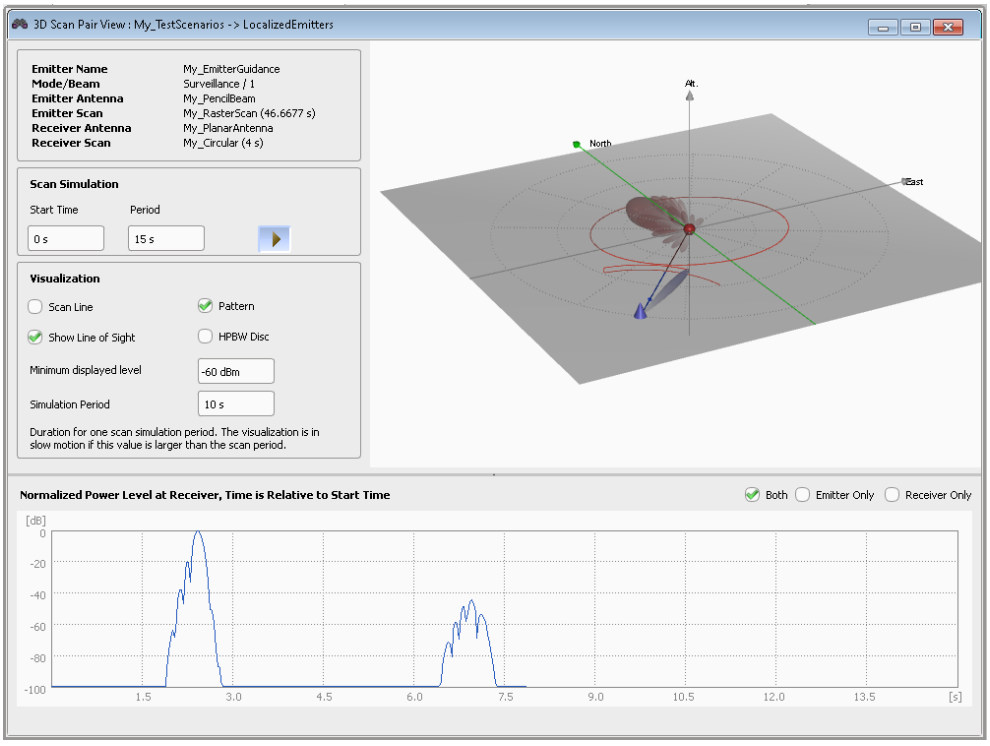
Pointing Direction
Defines the orientation of the antenna as "Elevation", "Bearing" and "Altitude".

Remote command:
SCENario:LOCalized:RECeiver:DIRection:ELEVation on page 364
SCENario:LOCalized:RECeiver:DIRection:BEARing on page 364
SCENario:LOCalized:RECeiver:ALTitude on page 366

13.1.3 3D Scan Pair View Settings

To access this dialog:

1. In a "Localized Emitters" scenario, select "Emitters > 2D".
2. In the 2D map, open the context menu of an "Emitter" and select "3D view with Receiver".



This dialog displays:

- a live plot of the normalized signal power level at the receiver
- a 3D view of the receiver and emitter antennas with their patterns and scans

Use the following setting to adjust the display:

Scan Simulation..... 206
Visualization..... 206
Both, Emitter only, Receiver only..... 206

Scan Simulation
Sets the time span visualized on the live plot as a start time and simulation period.

The "Period", i.e. the total duration the antenna scan simulation, determines the max. value on the x-axis. The visualization is relative to the selected "Start Time".

Use the "Pause" icon to stop the visualization.

Visualization
Defines the way the scan is represented:

"Scan Line" The scan is visualized by a line; the antenna pattern is disregarded.

"Pattern" The visualization corresponds to the current selected antenna pattern.

"Show Line of Sight"
Displays the line of sight (LOS) between the emitter and the receiver.

"HPBW" The antenna is represented by its HPBW.
See also "To visualize the signal on a 3D scan" on page 213.

"Minimum displayed level"
Defines the minimum displayed side lobes level.

"Simulation Period"
Sets the time it takes the animation to complete a scan.

Both, Emitter only, Receiver only
Defines the signal strength of which signal is visualized.

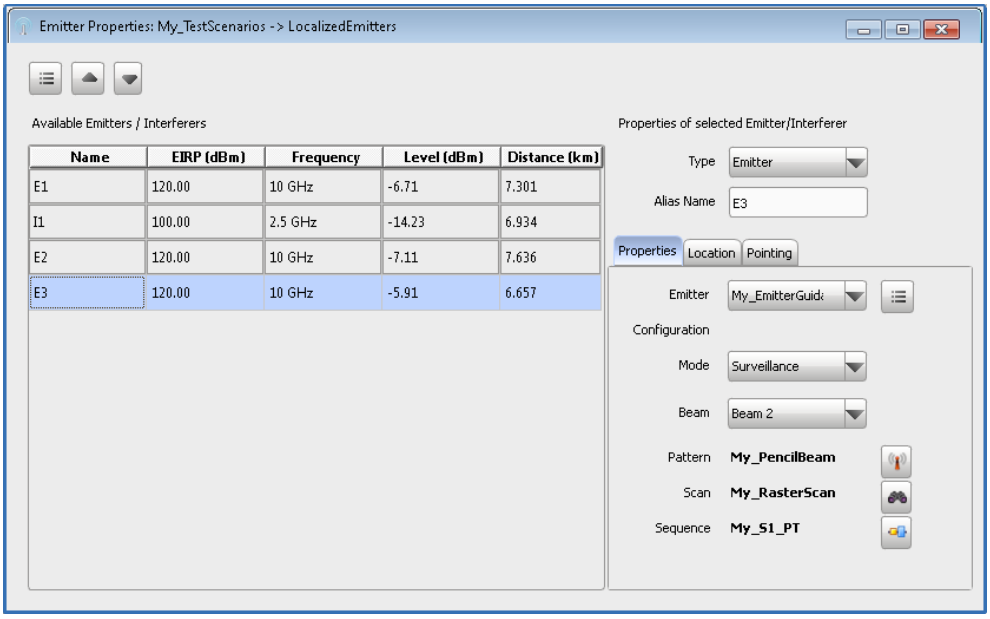
See also "To visualize the signal on a 3D scan" on page 213.

13.1.4 Lists with multiple emitters and interferers

To access this dialog:

1. In a "Localized Emitters" scenario, select "Emitters > 2D".
2. In the 2D map, open the context menu of an "Emitter" and select "Properties".

   The "Localized Emitters" scenario comprises several emitters and interferers. The available elements are described in a list form.



You can configure the following "Emitter/Interfere Properties":

List of Available Emitters/Interferers 207
Properties of the selected Emitter/Interferer 207
Location 208
Pointing Direction 208

List of Available Emitters/Interferers
Displays a list of emitters/interferers.

Use the standard functions in the context menu to add, reorder or remove items.

See also "To enable static emitters and place them on the 2D map" on page 211.

Remote command:
`SCENario:LOCalized:ADD` on page 279
`SCENario:LOCalized:SELect` on page 280
`SCENario:LOCalized:CURRent` on page 363
`SCENario:LOCalized:DELete` on page 281

Properties of the selected Emitter/Interferer
Each emitter/interferer is described with the following parameters:

"Type" Defines whether an emitter or an interferer is configured.

"Alias Name" Enters an alias name.

"Emitter/Interferer Name"
Selects an emitter/interferer.

"EIRP" Sets the EIRP of the interferer.

"Frequency" Sets the frequency of the interferer.

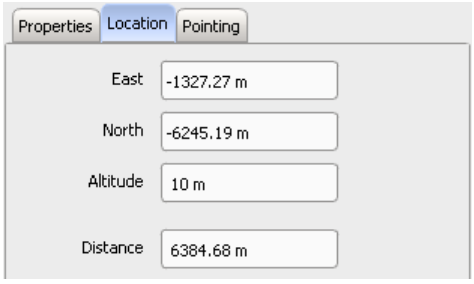
"Mode" Selects the mode the emitter is working in.

"Beam" Sets the number of current used beam.

"Antenna Pattern, Scan, Sequence"
Displays the current selected antenna pattern, antenna scan and sequence.
To change any of them, select the corresponding icon.

Remote command:
SCENario:LOCalized:TYPE on page 367
SCENario:LOCalized:ALIas on page 363
SCENario:LOCalized:EMITter on page 364
SCENario:LOCalized:EMITter:MODE on page 365
SCENario:LOCalized:EMITter:MODE:BEAM on page 365
SCENario:LOCalized:WAVeform on page 367
SCENario:LOCalized:WAVeform:EIRP on page 367
SCENario:LOCalized:WAVeform:FREQuency on page 368
SCENario:LOCalized:WAVeform:ANTenna on page 366
SCENario:LOCalized:WAVeform:SCAN on page 367

Location
The emitters and interferes on the 2D map are static elements with configurable coordinates.



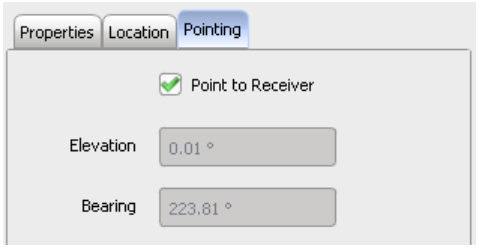
To change the location of an element, drag this element on the 2D map.

Use the "Location" parameters to define the position more precisely.

See also "To enable static emitters and place them on the 2D map" on page 211.

Remote command:
SCENario:LOCalized:LOCation:EAST on page 366
SCENario:LOCalized:LOCation:NORTh on page 366
SCENario:LOCalized:LOCation:ALTitude on page 366
SCENario:LOCalized:DISTance? on page 365

Pointing Direction



Defines the orientation of the antennas as "Elevation" and "Bearing".

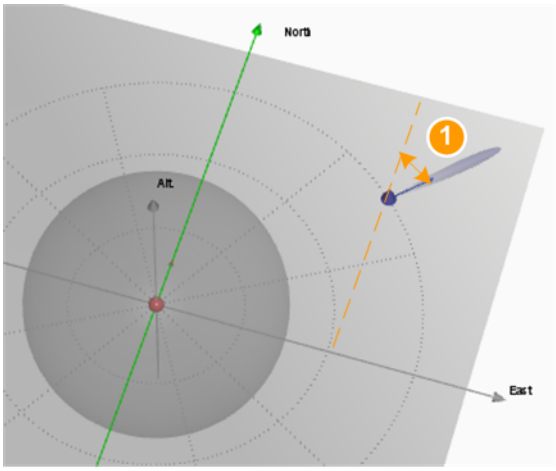


Fig. 13-3: Bearing: understanding the displayed information

1 = Bearing is the angle between the antenna's boresight and North

If "Point to Receiver" is enabled, the antenna is automatically turned in the direction of the receiver.

Remote command:
`SCENario:LOCalized:DIRection:TRACk` on page 366
`SCENario:LOCalized:DIRection:ELEVation` on page 364
`SCENario:LOCalized:DIRection:BEARing` on page 364

13.2 How to Create Scenarios with Emitters, Interferers and a Receiver

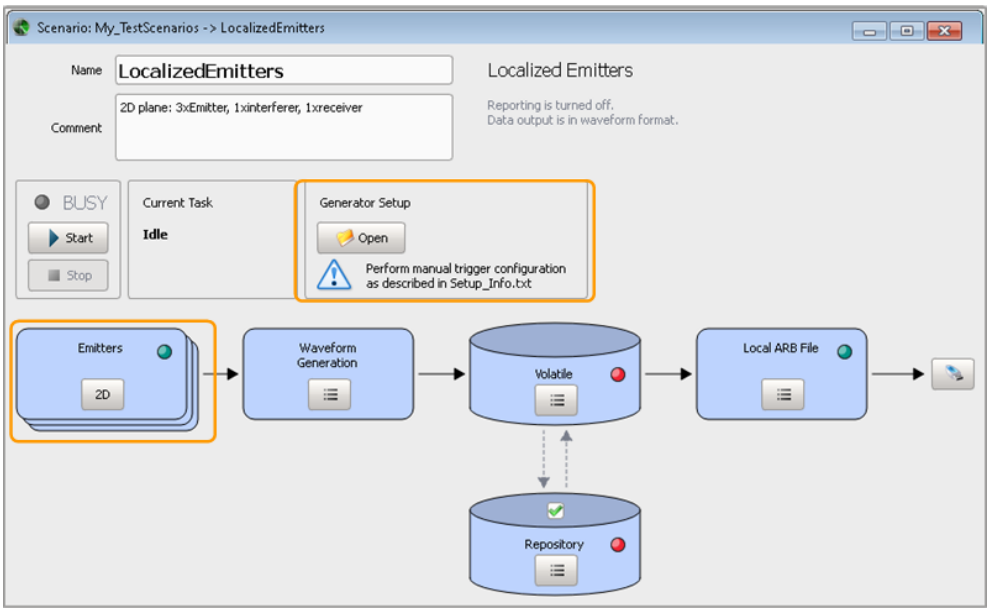
If you test situation requires to test the receiver's ability to correctly detect the sum signal from different static emitters and interferers, you can use "Localized Emitter" scenario. In this dedicated scenario, you configure the signal of one ore more emitters and optional one or more interferers that would be received by a receiver with defined characteristics. You can also configure the receiver and change its position in the scanning beam of the emitters.

To explain the principle, the following examples create a complex scenario with two emitters and one interferer that are located at a given distance to the receiver; the receiver characteristics can also be configured:

- "Open a suitable scenario" on page 210
- "To add an existing waveform as an interferer to the scenario" on page 210
- "To enable static emitters and place them on the 2D map" on page 211
- "To configure the receiver settings" on page 213
- "To visualize the signal on a 3D scan" on page 213
- "To map the generated signals to the signal generator" on page 214

Open a suitable scenario

1. Create a new "Scenario > Localized Emitters".



2. In the block diagram, select "Emitter > 2D".

   The "2D Localized Emitters" dialog opens and displays a 2D view of the receiver and currently configured emitters and interferes, together with their main characteristics.
   See figure 13-4.

3. Enable and configure emitters, interferers and the receiver, see:
   - "To enable static emitters and place them on the 2D map" on page 211
   - "To add an existing waveform as an interferer to the scenario" on page 210
   - "To configure the receiver settings" on page 213

4. Use the context menu to:
   a) display the 3D antenna patterns and scans
   b) to add/remove an emitter or an interferer

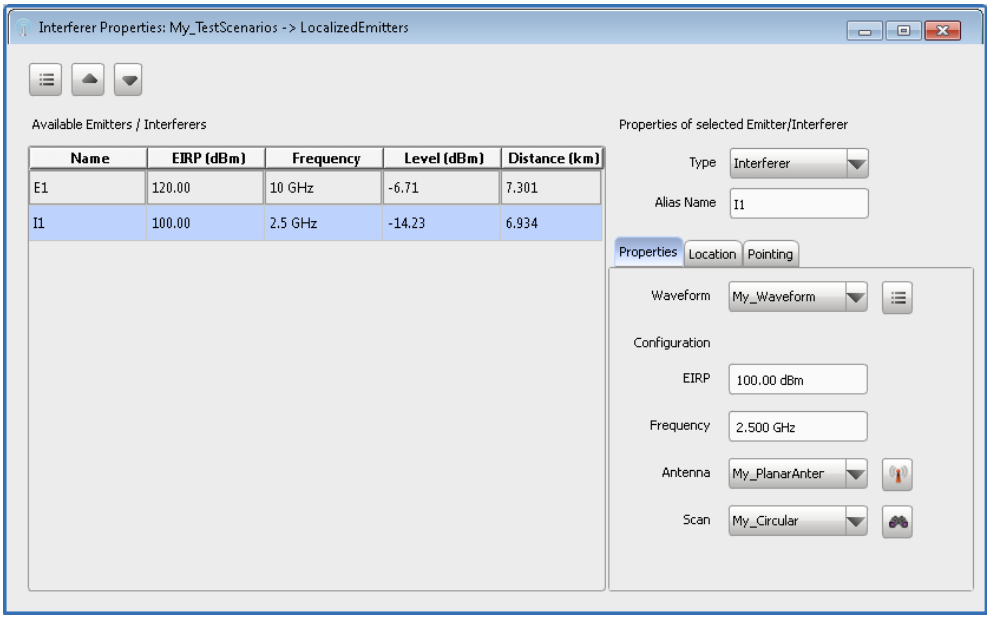
5. Use the mouse wheel to change the scale of the 2D diagram, i.e. to change the distance between the receiver and the emitters/interferers.

   For description of the provided settings, see chapter 13.1, "Related Settings", on page 202.

To add an existing waveform as an interferer to the scenario

1. To add an interferer:
   a) select an waveform in the repository tree and drag&drop it into the 2D diagram
      or
   b) use the context menu in the 2D diagram.

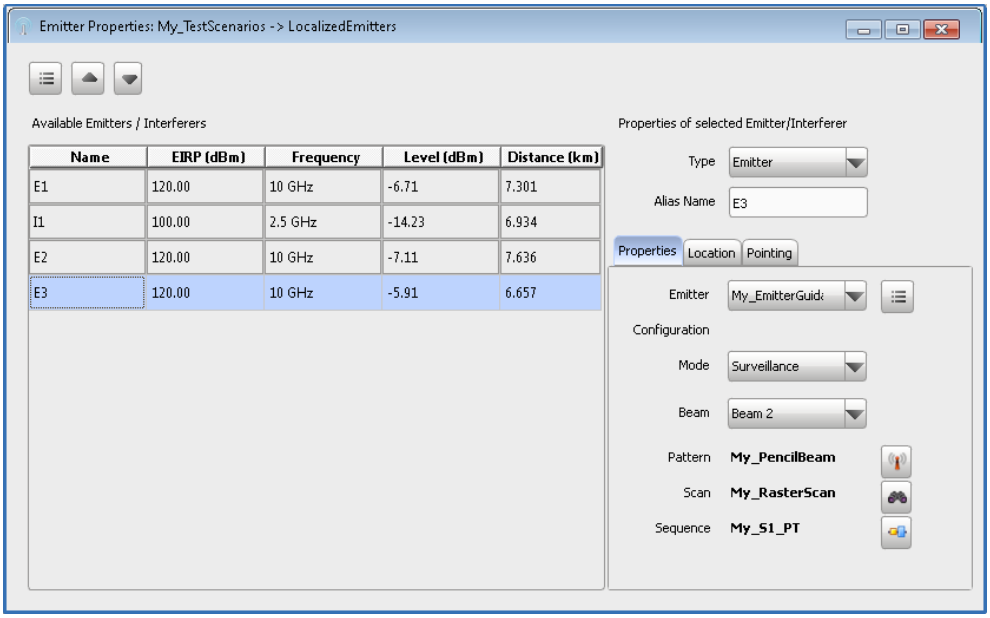
The "Emitter/Interferer Properties" dialog opens and lists all emitter and interferes configured in the scenario.



2. Configure the interferers's properties, e.g.:
   a) assign one of the existing antenna patterns or create a new one
   b) use the "Location" parameters to define the emitter's position more precisely
   c) use the "Pointing Direction" parameters to change the orientation of the antenna

To enable static emitters and place them on the 2D map

1. To add an emitter, drag&drop an emitter it into the 2D diagram.
2. Configure the emitter's properties.



The 2D diagram displays the configuration on a map with North/East coordinates.

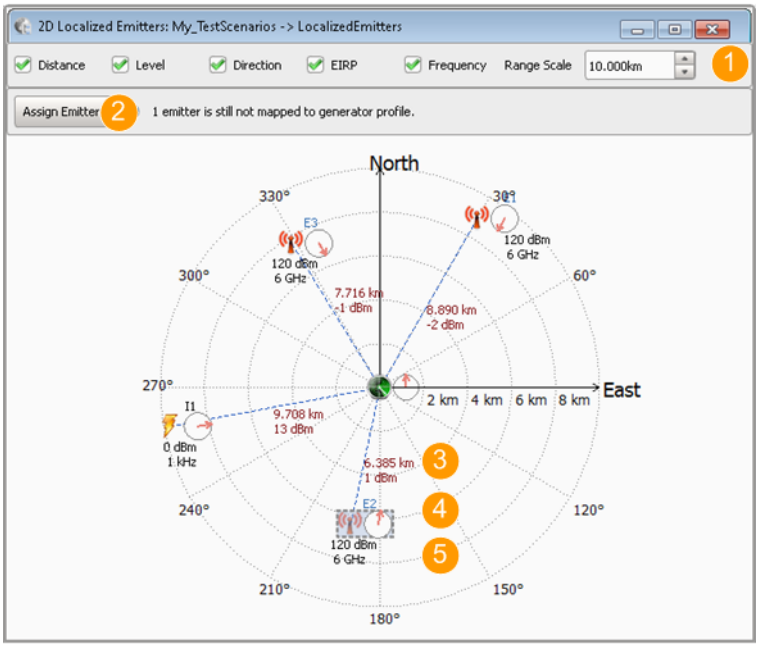
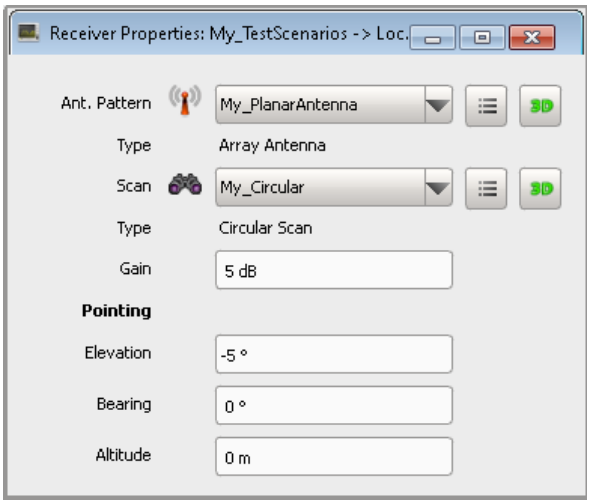


Fig. 13-4: 2D view: understanding the displayed information

1 = Display options; the "Range Scale" for example defines the radius of the displayed 2D map
2 = Access the "Assign Signal Source to Generator Profiles" dialog and indicates the assignment status
3 = Distance to the receiver and strength of the received signal
4 = "Point to Receiver = On" (antenna is automatically turned in the direction of the receiver)
5 = Emitter (or interferer) indicated with its alias name, EIRP and transmission frequency

To configure the receiver settings

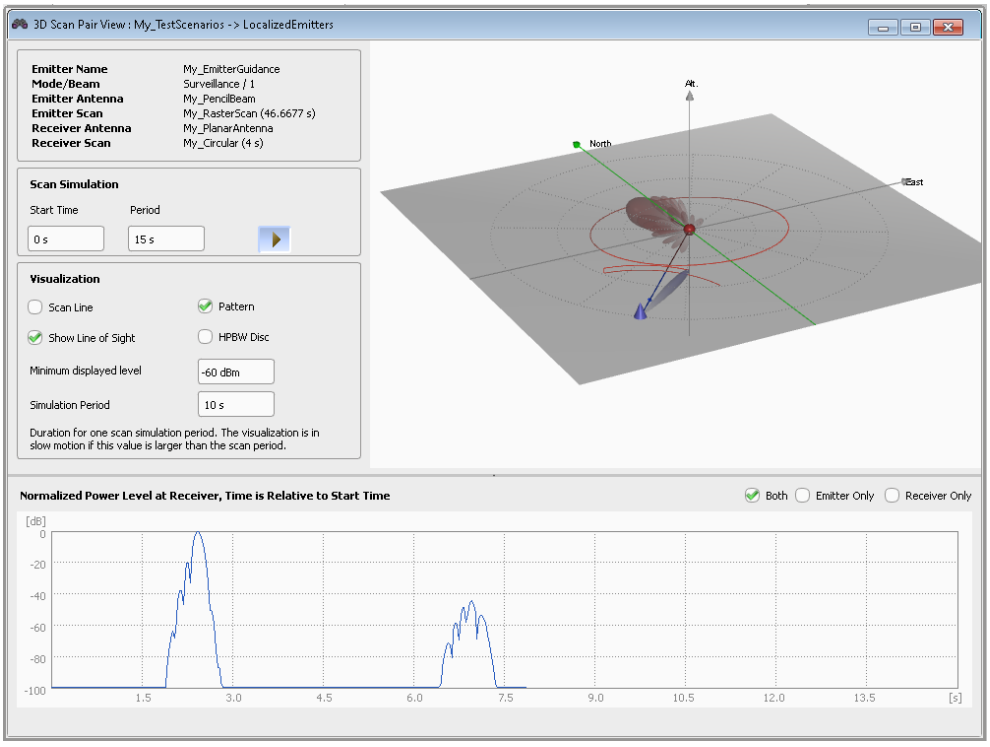
► In the 2D diagram, open the context menu of the receiver and select "Properties".



For description of the provided settings, see chapter 13.1.2, "Receiver Settings", on page 204.

To visualize the signal on a 3D scan

1. In the 2D diagram, open the context menu of an emitter/interferer. e.g. "E2" and select "3D View with Receiver".
2. Select "Normalized power Level at Receiver > Both > On" to display the resulting signal.



3. To display only the signal at the receiver, enable "Emitter Only".

   The calculation assumes a receiver with an omnidirectional antenna pattern.

4. Compare the displayed signal with the signal received by a static receiver in a "Single Emitter" scenario with the same emitter "E2".
   See figure 11-3.

   For description of the provided settings, see chapter 13.1.3, "3D Scan Pair View Settings", on page 205.

To map the generated signals to the signal generator

In a "Scenario Type > Localized Emitters", you have to select the signal of which emitter/interferer is currently transmitted to which signal generator.

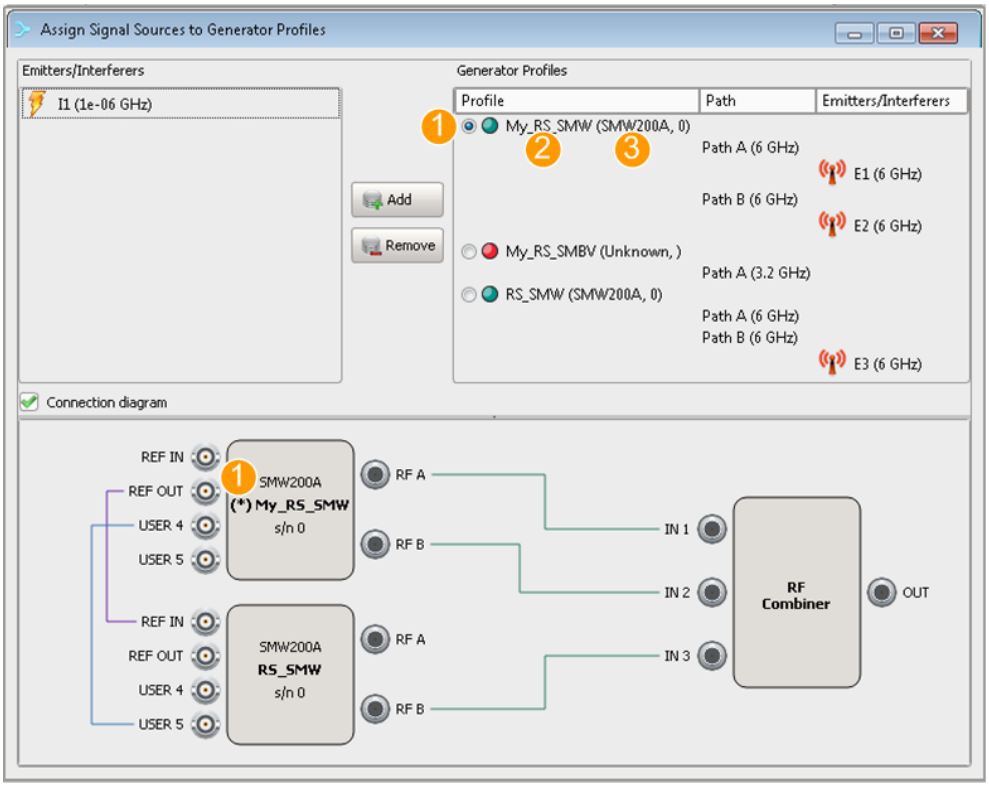
You can assign more then one signals to each of the instrument paths.

1. In the "Scenario" dialog, perform one of the following:

   a) select "Upload to VSG > Assign Emitters"
   b) select "Emitters > 2D > Assign Emitters"

   The "Assign Signal Source to Generator Profiles" dialog opens.

2. Drag&Drop an Emitter/Interferer to assign its signal to a generator profiles.



1 = Master instrument
2 = Generator profile (see chapter 14.2, "Generator Profile Settings", on page 219)
3 = Physical instrument and serial number

3. Enable "Connection diagram > On".

The diagram visualizes the required connections. The **(*)** sign indicates the signal generator that acts as a master instrument.

4. Provide all other instruments with the reference frequency and the trigger signal of the master signal generator.
   See also "To create, load and play the waveforms of a 2D scenario automatically" on page 255.

14 Creating Generator Profiles and Configuring the Instruments

Files generated by the R&S Pulse Sequencer can be transferred to and played by remotely connected signal generators.

The connection between the software and the signal generator is configured with the following steps:

- Create a **generator profile**
  The generator profile is a logical placeholder for a physical instrument. It describes the minimum requirements that must be fulfilled so that the signal can be generated.
  When a new empty repository is created, one default generator profile is created, too.
- Configure a **physical instruments**
  Generator profiles are assigned to physical instruments.
- Assign a generator profile to the scenario.

14.1 Instrument Configuration Settings

To access these settings:

► Select "Menu bar > Configure > Instruments".

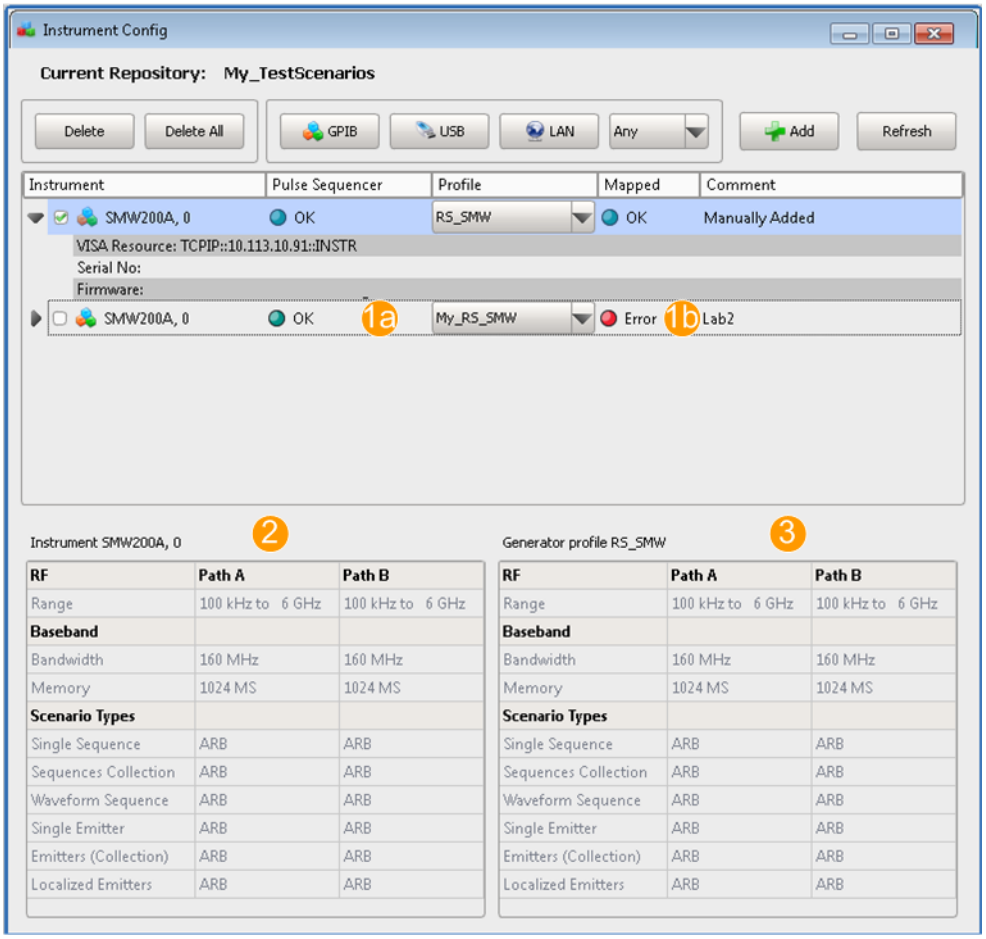


Fig. 14-1: Instrument Configuration: understanding the displayed information

1a, 1b = status messages; "Mapped > Error" indicates insufficient instrument's configuration for the current profile
2 = lists the retrieved instrument configuration
3 = lists the profile capabilities, see chapter 14.3, "How to Create Generator Profiles and Configure the Connected Instruments", on page 221

The following settings are available:

Repository....218
Delete, Delete All....218
Scan USB/GPIB/LAN, Any....218
Add....218
Refresh....218
List of connected instruments....218
Device Capabilities, Profile Capabilities....219

Repository
Indicates the current repository.

Delete, Delete All
Standard edit functions to handle the list of instruments.

Remote command:
`INSTrument:DELete` on page 281
`INSTrument:CLEar` on page 281

Scan USB/GPIB/LAN, Any
Scans the selected local interface for connected instruments. To limit the scan time, select the instrument's type, e.g. "SMW200A".

The GPIB scan is limited to the first GPIB card that is installed (GPIB0).

Add
Adds an instrument manually.

If an instrument with the selected IP address/computer name or complete VISA resource string exists, it is added to the list of instruments.

Remote command:
`INSTrument:ADD` on page 308

Refresh
Standard edit function.

List of connected instruments
Displays information on all known and previously configured instruments, like the used remote control interface, firmware version and status information on the performed capabilities check.

- Select "Profile" to change the profile the connected instrument is compared to. See "To find connected instruments and assign the generator profiles to them" on page 222.
- To update an instrument on "Refresh", select its checkbox.
- Disable an instrument in the list, if it is temporarily unavailable but should not be deleted from the list.
  If you disable an instrument that is currently being used in one of the scenarios, this scenario is configured to store the calculated waveform as an ARB file.
  See also "Load ARB File" on page 59.
- Double click on an instrument to change the displayed "Comment".

Remote command:
`INSTrument:SELect` on page 280
`INSTrument:MAP` on page 309
`INSTrument:COUNt?` on page 280
`INSTrument:FIRMware?` on page 309
`INSTrument:STATus?` on page 309

Device Capabilities, Profile Capabilities
A side-to-side overview of the provided and required capabilities.

Remote command:
`INSTrument:CAPabilities?` on page 308

14.2 Generator Profile Settings

To access these settings:

► Select "Repository Tree > Generator Profiles > New" or open the automatically created default profile.

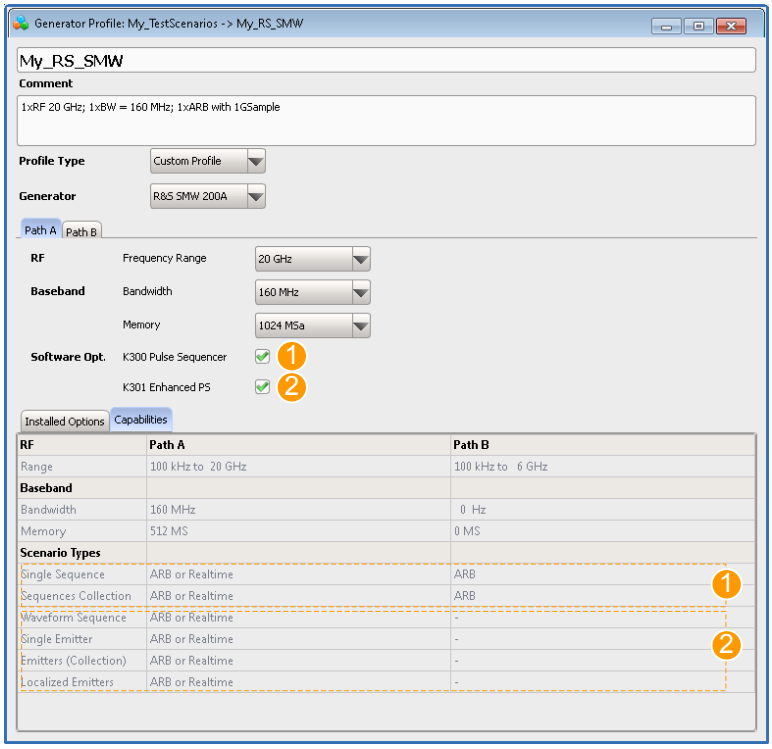


Fig. 14-2: Generator Profiles: understanding the displayed information

1, 2 = Available options and their features

The following settings are available:

Name....220
Comment....220
Profile Type....220
Generator....220
List of Available R&S Signal Generators....220
RF Frequency Range....220
Baseband....221

Software Options.....221
Installed Options..... 221
Capabilities.....221

Name
Enter the name of the generator's profile.

Remote command:
`GENerator:NAME` on page 277
`GENerator:CREate` on page 277
`GENerator:CATalog?` on page 276
`GENerator:SELect` on page 277
`GENerator:REMove` on page 278

Comment
Enter a short description.

Remote command:
`GENerator:COMMent` on page 278

Profile Type
Defines the way the generator's profile is determined.

The hardware characteristics of the selected generator determines the way the generated waveform file is processed. In particular, the available ARB bandwidth limits the waveform length.

See also chapter 19, "Playing the Generated Waveform Files", on page 247.

"Custom"
The instrument configuration is selected manually.
See "To create a new custom generator profile" on page 221

"Connected"
The instrument's configuration is retrieved automatically from the connected instrument.
See "To retrieve the profile of the connected generator" on page 224

Generator
Selects the type of the signal generator form the list of supported instruments.

Remote command:
`GENerator:TYPE` on page 307

List of Available R&S Signal Generators
With "Profile Type > Connected", lists all instruments that are configured in the Instrument Configuration dialog.

To create the profile of a particular instrument, use the "Create Profile" function.

RF Frequency Range
Sets the maximum RF frequency, per RF output.

Remote command:
see `GENerator:OPTion?` on page 308

Baseband
Sets the baseband related parameters:

"Bandwidth" Sets the maximum supported baseband bandwidth.

"Memory Extension"
Sets the total available ARB memory.

Remote command:
see `GENerator:OPTion?` on page 308

Software Options
Selects the installed/required pulse sequence options, e.g. "K300 Pulse Sequencer" or "K301 Enhanced PS".

The Capabilities is updated and displays information on:
- which scenario types are supported with this particular combination of features
- whether the generated waveform file can be processed as an ARB file.

Installed Options
An overview of the installed instruments options.

Remote command:
`GENerator:OPTion?` on page 308

Capabilities
An overview of the generator capabilities regarding supported scenario types and processing of waveforms and multi segment files.

Remote command:
`GENerator:CAPabilities` on page 308

14.3 How to Create Generator Profiles and Configure the Connected Instruments

Configure the remotely connected instrument connections first, see:
- "To create a new custom generator profile" on page 221
- "To find connected instruments and assign the generator profiles to them" on page 222
- "To retrieve the profile of the connected generator" on page 224
- "To assign a connected instrument to a scenario" on page 225

To create a new custom generator profile

1. Select "Repository Tree > Generator Profiles > New".
2. Enter a name and add a comment.
3. Select "Profile Type > Custom Profile".
4. Set the type of used signal generator, e.g. "Generator > R&S SMW200A".

5. Configure the instruments settings, like max. RF, baseband Bandwidth, installed or required features etc.

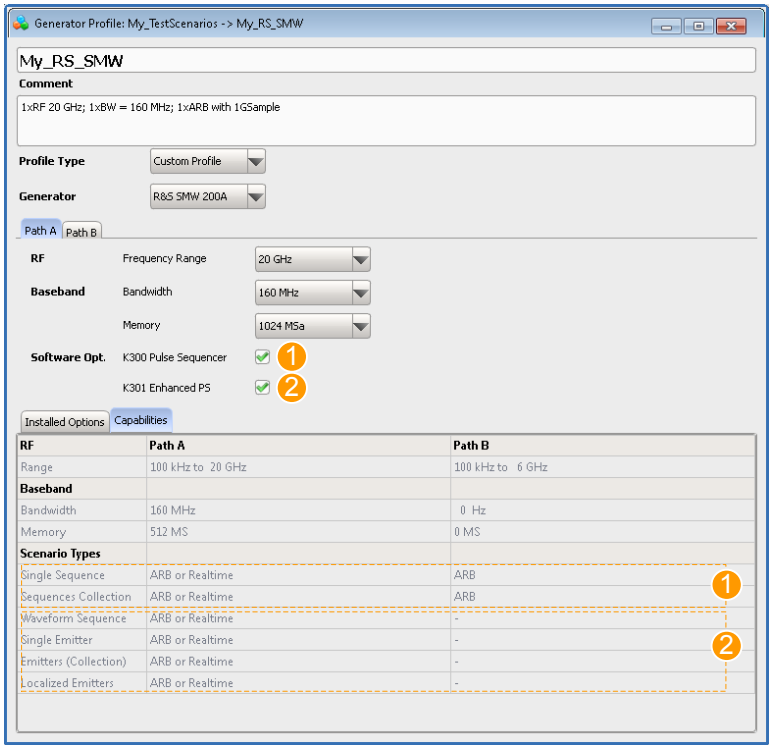


Fig. 14-3: Generator Profiles: understanding the displayed information

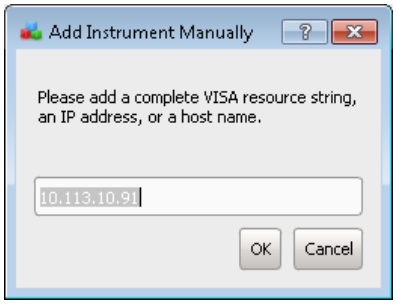
1, 2 = Available options and their features

The list of "Installed Options" and the "Capabilities" are updated.
For description if the provided settings, see chapter 14.2, "Generator Profile Settings", on page 219.

To find connected instruments and assign the generator profiles to them

1. Connect a signal generator and the PC the R&S Pulse Sequencer software is running on to the LAN, see figure 4-1.
2. Select "Menu bar > Configure > Instruments".
3. To add an instrument, perform one of the following:

a) Select "Add", and enter the IP address, the computer name or the complete VISA resource string of the connected instrument and select "Ok".



See description R&S®SMW200A Getting Started.

b) Select "Scan LAN".

The LAN is scanned for connected instruments. If suitable instruments are found, the dialog shows them, together with information on the their capabilities.

4. Select "Profile" and select a generator profile from the list.

The instrument capabilities are compared to the profile requirements in regard of processing the current configured scenario and pulse sequence.
Note: The connected instrument requires at least one option R&S SMW/SMBV/SGT-K300 to play ARB files generated with R&S Pulse Sequencer.

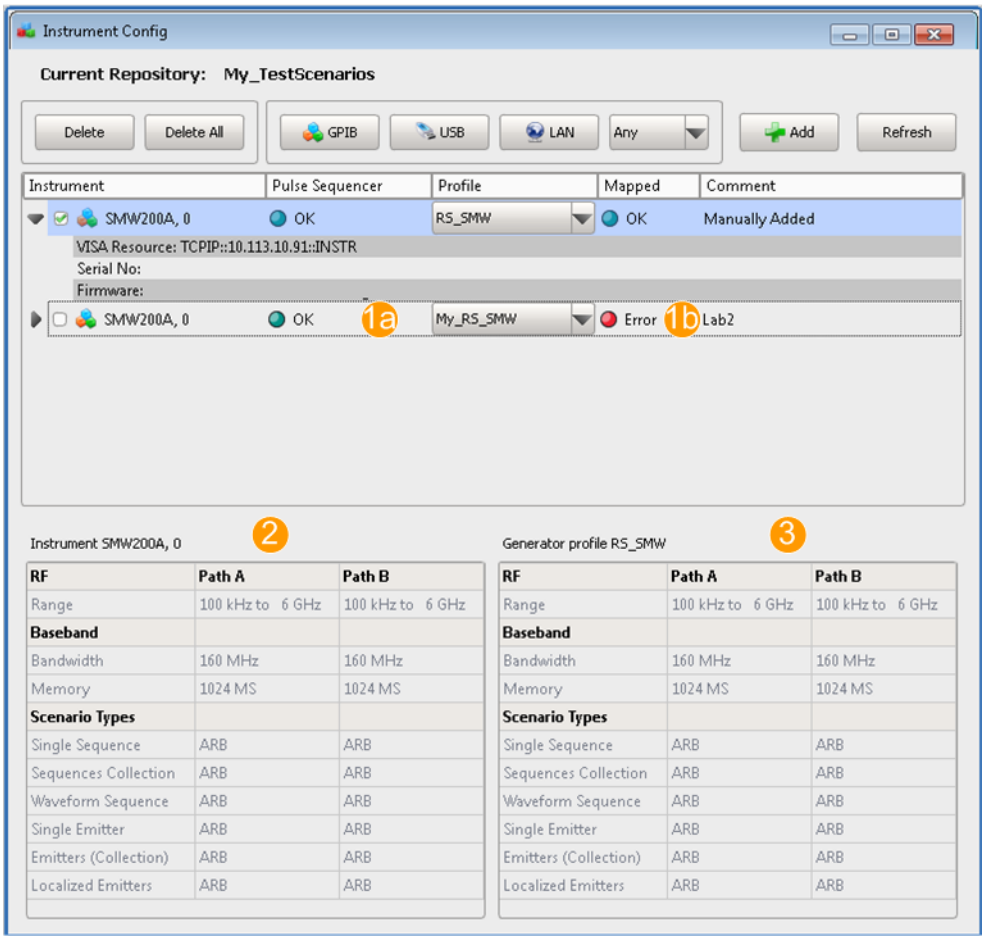


Fig. 14-4: Instrument Configuration: understanding the displayed information

1a, 1b = Status messages; "Mapped > Error" indicates insufficient instrument's configuration for the current profile
2 = List of the retrieved instrument configuration
3 = List of the profile capabilities, see chapter 14.3, "How to Create Generator Profiles and Configure the Connected Instruments", on page 221

5. Observe the status indication.
   - **Green LED**/"OK": indicates a suitable instrument's configuration
   - **Yellow LED**/"N/A": indicates an unknown instrument or unsuccessful mapping
   - **Red LED**/"Error": indicates that the requirements are not matched, e.g. missing option, insufficient amount of memory, etc.

   For description if the provided settings, see chapter 14.2, "Generator Profile Settings", on page 219.

To retrieve the profile of the connected generator

1. Create a new "Generator Profiles", see "To create a new custom generator profile" on page 221.
2. In the "Generator Profiles" dialog, select "Profile Type > Connected".
3. Select "Generator" and select the generator type.

4. In the list of the "Available Signal Generators", select an instrument and select "Create Profile".

   The created profile match the capabilities of the connected instrument.

5. In the "Instrument Configuration" dialog, assign this profile to the connected signal generator.
   See "To find connected instruments and assign the generator profiles to them" on page 222.

To assign a connected instrument to a scenario

1. Open an existing scenario, e.g. "SimplePulseTrain".
2. Select "Upload to VSG > context menu" and select:
   a) "Target > Generator"
   b) "Select" and select a generator's profile, e.g. "R&S_SMW".

   The displayed information confirms the selected profile.
3. Observe the status indication.
   - **Green LED**: indicates that the selected profile is suitable for the current scenario;
     If all other prerequisites are fulfilled, the "Start" function is active.
   - **Yellow LED**: indicates insufficient capabilities
   - **Red LED**: indicates that no profile is assigned

   See chapter 19, "Playing the Generated Waveform Files", on page 247.

15 Defining the Data Source

The R&S Pulse Sequencer provides a wide range of internally defined data sources, such as patterns or PRBS generators. Data sources are available globally within the project. Once created, they can be used in all pulses. Each pulse retrieves its data individually.

Data is retrieved cyclically, starting from the first list entry. Once all bits are used, the data is retrieved form the second one. When the list end is reached, the data is retrieved from the first entry.

If the data entry contains fewer bits than the required, the data bits are repeated cyclically.

Example: Data source processing

Configure the data source as shown on figure 15-1.

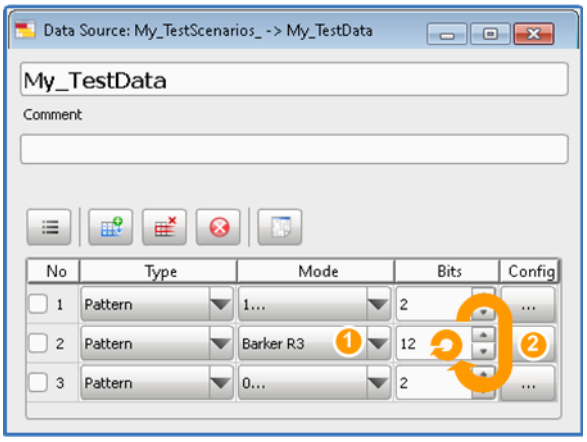


Fig. 15-1: Data Source: understanding the displayed information

1 = Barker R3 pattern = 110; required are 12 bits, i.e. the pattern is repeated 4 times
2 = If more than 16 bits are required, the 16 bits are repeated

Use the Preview Data Source to visualize the content.



1 = Displayed length is 16 bytes
2 = Configured 16 bits (2 bytes), repeated 8 times
3 = Content of each 2 bytes: 2x"Pattern = 1" 4x"Pattern = Barker R3" 2x"Pattern = 0"

Supported types of data source

The R&S Pulse Sequencer provides the following types of data sources:

- *Pattern*
  - Simple data patterns such as binary 0 ("0...") strings or 1 ("1...") strings

  - Variable bit strings ("1010") with alternating 0 and 1 and a maximum length of 999 bits
  - Different Barker codes: Barker R3, Barker R4a, etc.

- *User*
  A user-defined sequence of 0 and 1 bits
  The following rules apply for user defined data source:
  - Accepted are 0 and 1 as well as numbers in hexadecimal format
  - Comments can be enclosed in slashes (/)
  - Blank characters are ignored
  - The sequence #x starts a hexadecimal input for the remainder of the line
  - A new line turns hexadecimal input and comments off
  - Ones and zeroes are evaluated as single bits
  - Strings can be enclosed in quotation marks (")
- *PRBS* (Pseudo-Random Binary Sequence)
  The PRBS generators deliver pseudo-random binary sequences of differing length and duration. They are known as maximum length sequences, and are generated with the aid of ring shift registers with feedback points determined by the polynomial.
  The pseudo-random sequence from a PRBS generator is uniquely defined by the register number and the feedback, see table 15-1.
  Example:
  By way of example, the diagram below shows a 9-bit generator with feedback to registers 4 and 0 (output). The generated serial data is converted internally, e.g 2 Bit/Symbol in case of QPSK.

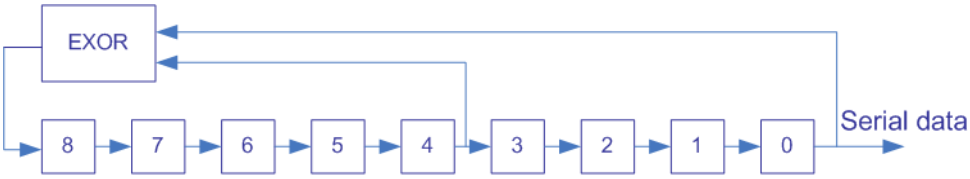


Fig. 15-2: A 9-bit PRBS generator

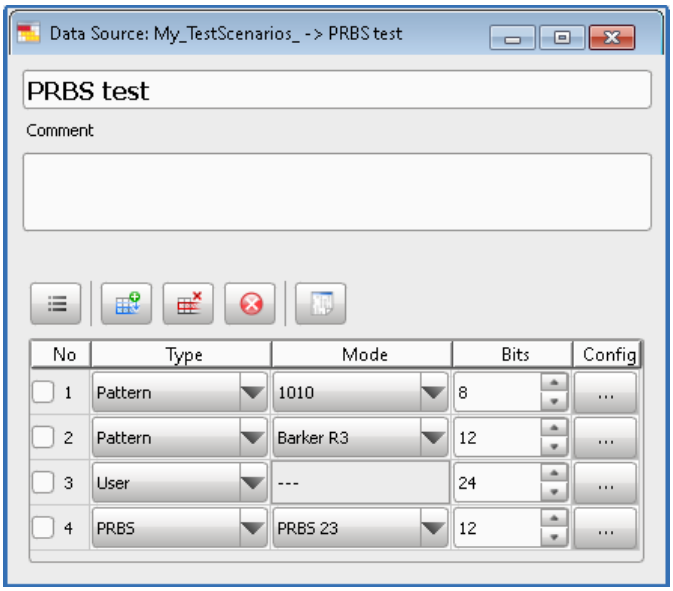
Table 15-1: Overview of PRBS generators

PRBS generator	Length in bits	Feedback to
7-bit	2^7 -1 = 127	Registers 1, 0
9-bit	2^9 -1 = 511	Registers 4, 0
11-bit	2^{11} -1 = 2047	Registers 2, 0
15-bit	2^{15} -1 = 32767	Registers 1, 0
16-bit	2^{16} -1 = 65535	Registers 5, 3, 2, 0
20-bit	2^{20} -1 = 1048575	Registers 3, 0
21-bit	2^{21} -1 = 2097151	Registers 2, 0
23-bit	2^{23} -1 = 8388607	Registers 5, 0

15.1 Data Sources Settings

To access these settings:

► Select "Repository Tree > Data Source > New".



Provides are the following settings:

Data Source Name....228
Comment....228
Data Source Table....229
Shift register....229
User Data....230
Preview Data Source....230

Data Source Name
Enter a name for the data source.

Remote command:
`DSRC:NAME` on page 277
`DSRC:CREate` on page 276
`DSRC:CATalog?` on page 276
`DSRC:SELect` on page 277
`DSRC:REMove` on page 278
`DSRC:CLEar` on page 281

Comment
Enter a short description.

Remote command:
`DSRC:COMMent` on page 278

Data Source Table
A table with one or more item that described the content of the data source.

"Item Select" Selects an item for performing standard tasks, like inserting an item above the current one, deleting an item, etc.

Remote command:
`DSRC:ITEM:SELect` on page 280
`DSRC:ITEM:ADD` on page 279
`DSRC:ITEM:DELete` on page 281

"Data Source Type, Data Source Mode"
Sets the type of the data and the data content for the current item, see "Supported types of data source" on page 226.

See also:

- chapter 15.2, "How to Configure the Bit Stream Used by the MOP", on page 230
- example "Data source processing" on page 226

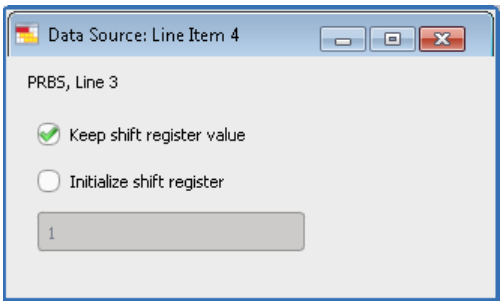
Remote command:
`DSRC:ITEM:TYPE` on page 301
`DSRC:ITEM:DATA` on page 299
`DSRC:ITEM:PATTern` on page 300
`DSRC:ITEM:PRBS:TYPE` on page 300

"Bits" Sets the length of the current item in bits.

Remote command:
`DSRC:ITEM:BITS` on page 299

"Config" Accesses a dialog with additional settings, see: "Shift register" on page 229

Shift register
A PRBS sequence is defined by a length and an register shift number, see table 15-1



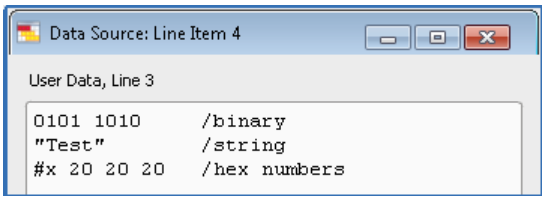
"Keep shift register value"
Use the default values.

"Initialize shift register"
Customize the shift register.

"Initialization value"
Set a new initialization value.

Remote command:
`DSRC:ITEM:PRBS:INIT` on page 300
`DSRC:ITEM:PRBS:INIT:VALue` on page 300

User Data
For "Data Source > Type > User", you can define your own data pattern, see "Supported types of data source" on page 226.



Preview Data Source
Displays the data content.

See also example "Data source processing" on page 226.

"Refresh" Perform refresh to display the current data content.

"Start Byte, Length"
Sets the data bits currently displayed.

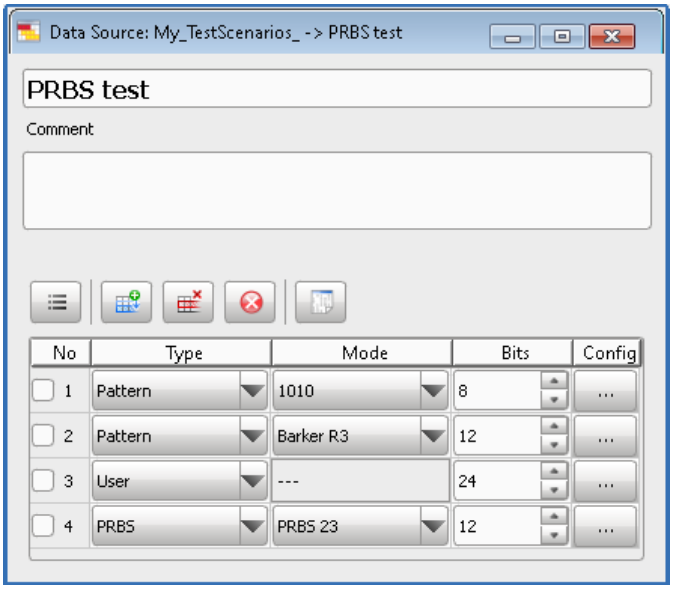
15.2 How to Configure the Bit Stream Used by the MOP

See:
- "To configure the data source" on page 230
- "To assign the data source to a MOP" on page 231

To configure the data source

To access the "Data Source" dialog:

1. Select "Repository Tree > Data Source > New".
2. Select the "Append new item" icon.
3. Enter a name and add a comment.
4. For each item, select the data type, the data mode and the data length in bits.



5. To view the data source content, select the "Preview" icon.

   Data is retrieved cyclically, starting from the first list entry, see example "Data source processing" on page 226.

To assign the data source to a MOP

1. In the repository tree, select "Pulse > PulseName e.g. My_TestPulse > MOP".
2. Select "MOP > Settings" and select a MOP that requires a data source, e.g. "MOP Type > FSK"
3. Select "Data Source > e.g. My_TestData".

16 Defining Complex Modulation Schemes and IPM Profiles

Custom modulation schemes and custom envelope shapes are defined by an external plugin. Plugins are Microsoft Windows DDL modules that contain the maths that is required for the envelope shaping and the modulation on pulse.

Some example plugins are provided with the software as binary and source code. These examples may serve as a starting point for own applications. Once imported, the software handles the plugins automatically.

Each plugin:

- have to provide a range of functions to identify itself and perform the calculations required for the modulation on pulse
  (see chapter A.3, "Plugin Programming API", on page 400)
- may register a set of configuration parameters that may be used as variables inside of the plugin
- can be used to report generation during the waveform creation process.

How to import and assign user defined plugins

See:

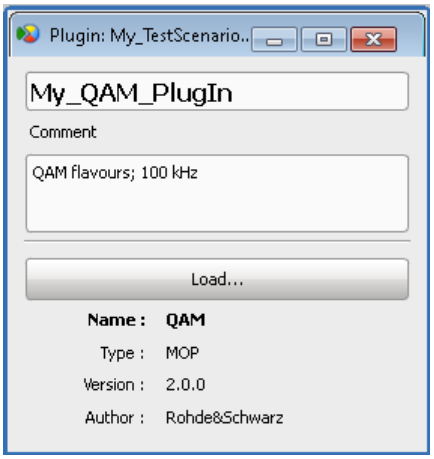
- "To import a plugin" on page 232
- "To assign the user defined modulation scheme to a pulse" on page 233
- "To assign the user defined IPM profile" on page 234

To import a plugin

1. In the repository tree, select "Plugins > New".
2. In the "Plugin" dialog, enter name and a description.
3. Select "Load", navigate to the `*.dll` file and select it.

   List and loaded are only files containing modulation data.

   The software loads the `*.dll` file, retrieves information form it and displays it in the "Plugin" dialog.



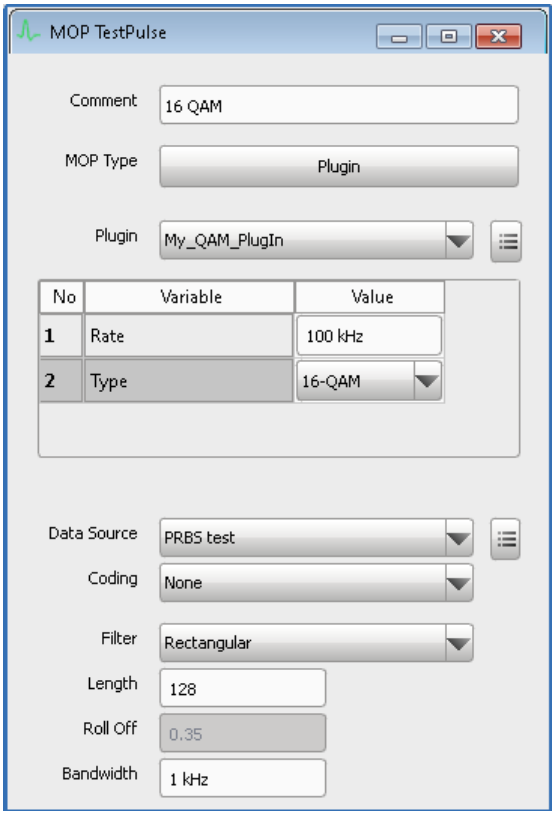
You can use the plugins as sources of custom modulation schemes, see "To assign the user defined modulation scheme to a pulse" on page 233.
See also chapter 7.2.6.7, "Plugins", on page 93.

Remote Commands:

- `PLUGin:CREate` on page 277
- `PLUGin:CATalog?` on page 276
- `PLUGin:NAME` on page 278
- `PLUGin:COMMent` on page 278
- `PLUGin:LOAD` on page 328
- `PLUGin:SELect` on page 277
- `PLUGin:REMove` on page 278
- `PLUGin:MODule:NAME?` on page 278
- `PLUGin:MODule:TYPE?` on page 328
- `PLUGin:MODule:VERSion?` on page 328
- `PLUGin:MODule:AUTHor?` on page 328
- `PLUGin:MODule:COMMent?` on page 328
- `PLUGin:MODule:DATA?` on page 328

To assign the user defined modulation scheme to a pulse

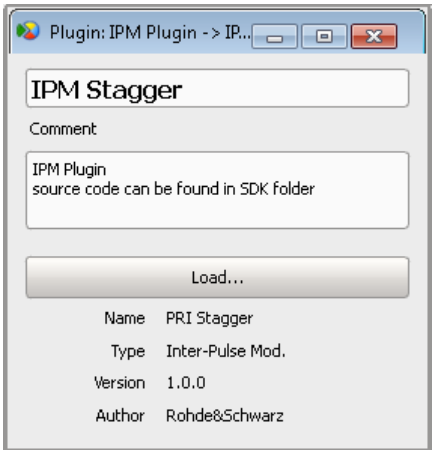
1. In the repository tree, select "Pulse > PulseName > MOP".
2. Select "MOP > Settings".
3. Select "MOP > MOP Type > Plugin".
4. Select "Plugin" and select a loaded user defined modulation scheme, e.g. `My_QAM_PlugIn`.



5. Select "Pulse > PulseName > MOP > Enable Modulation on Pulse (MOP) > On".
6. Use the Restrict modulation to a certain area of the pulse settings to determine the exact pulse part the modulation is applied to.

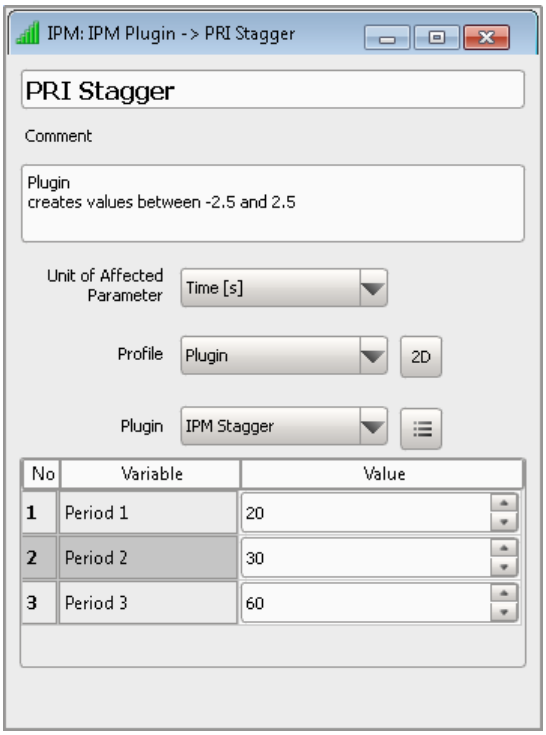
To assign the user defined IPM profile

1. Import a suitable plugin.



2. In the repository tree, select "Inter-Pulse Mods > IPM-Name".
3. Select "Profile > Plugin".

4. Select "Plugin" and select a loaded user defined IPM profile, e.g. `IPM Stagger`.



5. Select "2D" to visualize the IPM profile.
6. If required, change the values of the used variables and observe the effect on the "2D" diagram.

17 Defining and Enabling Marker Signals

Markers signals or markers are binary signals that are additionally generated and added to the generated signal. Markers are commonly used to synchronize a device under test (DUT) or a spectrum analyzer to the generated signal, for example to the beginning of a pulse. In the vector signal generator, generated marker signals can be routed to the corresponding output connectors.

For details, see description R&S®SMW200A User Manual.

In the R&S Pulse Sequencer, you define markers on three levels: on pulse basis, in the sequence or define global conditions like the start of a scenario.

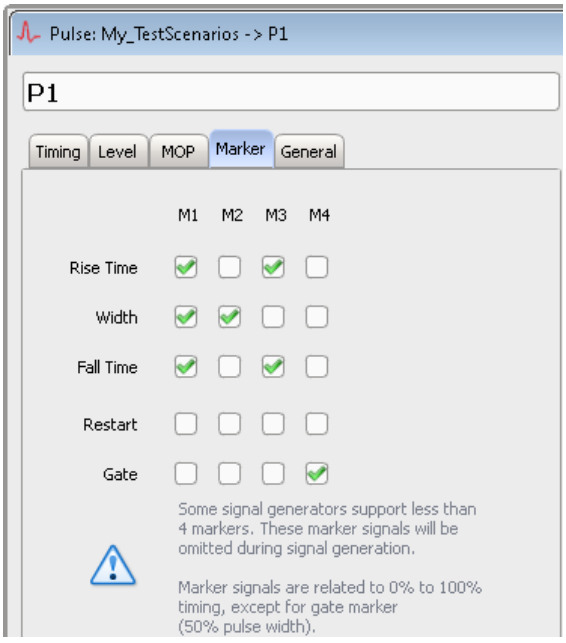
Pulse markers

With the provided settings, you can assign up to 4 markers to the different pulse phases, e.g. rise and fall time, pulse width. A restart marker indicates the first 10% of the entire PRI.

Marker information is directly added to the resulting waveform and the marker signal output is therefore synchronous with the waveform playback.

To access the pules marker settings:

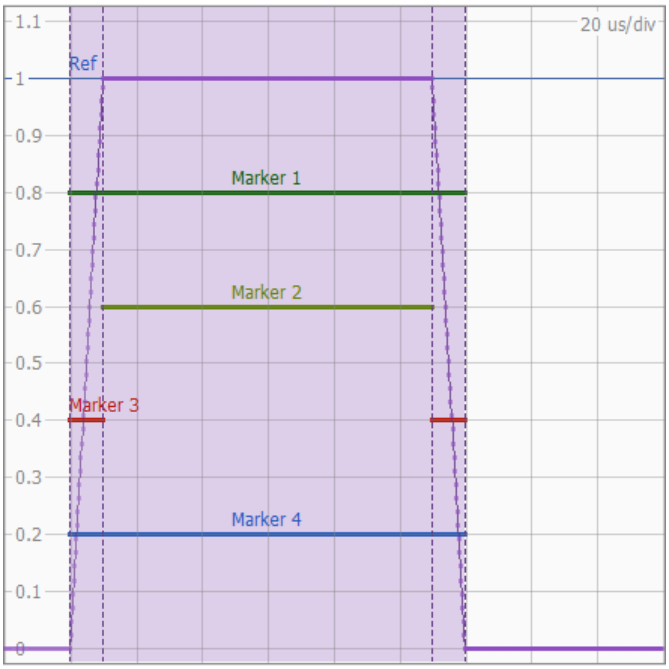
1. Select "Repository Tree > Pulse > Marker".
2. Enable Marker 1 ("M1") for Rise, Width and Fall Time.



   The Marker 1 is active for the entire number of samples, i.e. the marker is held high for the whole time from beginning of the rising edge to end of the falling edge.

3. Enable "M2 > Width", "M3 > Rise and Fall Time" and "M4 > Gate".

   The envelope graph marks the time periods during that the markers are held high.



Remote Commands:

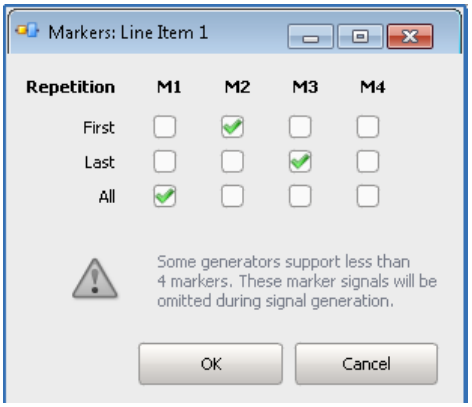
- `PULSe:MARKer:RISE` on page 324
- `PULSe:MARKer:WIDTh` on page 324
- `PULSe:MARKer:FALL` on page 324
- `PULSe:MARKer:AUTO` on page 324
- `PULSe:MARKer:GATE` on page 324

Sequence markers

If pulse markers are defined, you can also define markers on multiple repeating pulses and mark the first, last or all pulses.

To access the sequence marker settings:

1. Select "Repository Tree > Sequence > Sequence Description > Marker".
2. Select "M1 > All", "M2 > First", "M3 > Last".



The Marker 2 will be generate only for the first pulse out of all repetitions.

Remote Commands:

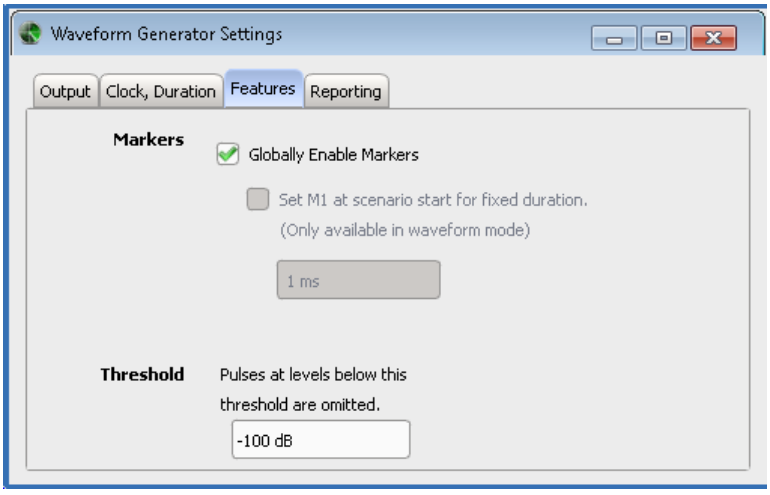
- `SEQuence:ITEM:MARKer:FIRSt` on page 324
- `SEQuence:ITEM:MARKer:LAST` on page 324
- `SEQuence:ITEM:MARKer:ALL` on page 324

Global markers

In scenarios, if the selected sequence contain markers, you can enable that they are considered by the generation of the output waveform file.

To access the waveform generator marker settings:

1. Select "Repository Tree > Scenario > Waveform Generator > Config".
2. Select "Features > Markers > Enable Markers Signals".



See also "Waveform Generation" on page 57.

3. Optionally, select "Set at scenario start for fixed duration > On" and set "Duration = 1 ms", to enable an additional global signal.

   This marker signal is output as marker 1. If other marker signals, e.g. pulse and sequence markers, are defined, all markers are summed.

How to visualize the configured markers

1. Open the "Waveform/Data Viewer", see "To access the "Waveform/Data View" dialog" on page 245.
2. Select "Marker > On".

   The "Waveform/Data Viewer" evaluates the marker information and displays the marker traces M1 to M4.

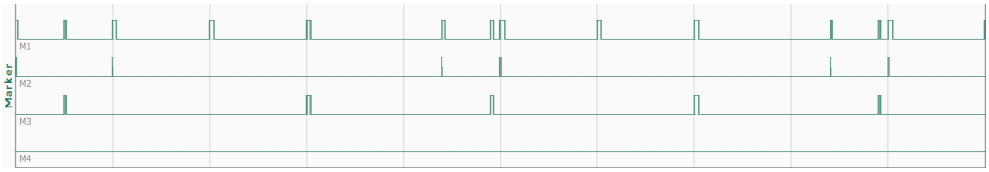


Fig. 17-1: Example: Waveform View > Marker traces

M1 = All
M2 = First
M3 = Last

18 Visualizing and Analyzing Waveforms

Waveform generated with the software or imported to the repository can be visualized with the built-in "Waveform/Data View" function. The R&S Pulse Sequencer reads in all a selected waveform, evaluates it and displays the I and Q data, the spectrum and the constellation diagram of a signal section or of the entire waveform. The "Waveform/ Data View" is only available if a waveform was created successfully.

Imported waveforms can also be evaluated to retrieve information on the reference signal level.

18.1 Waveform and Data View Settings

The waveform viewer reads the entire waveform data, evaluates it and visualizes the signal in three sections: The time domain view, the spectrum/waterfall and the I/Q constellation plot.

To access these settings:

► In the "Scenario" dialog, select "Volatile > View".

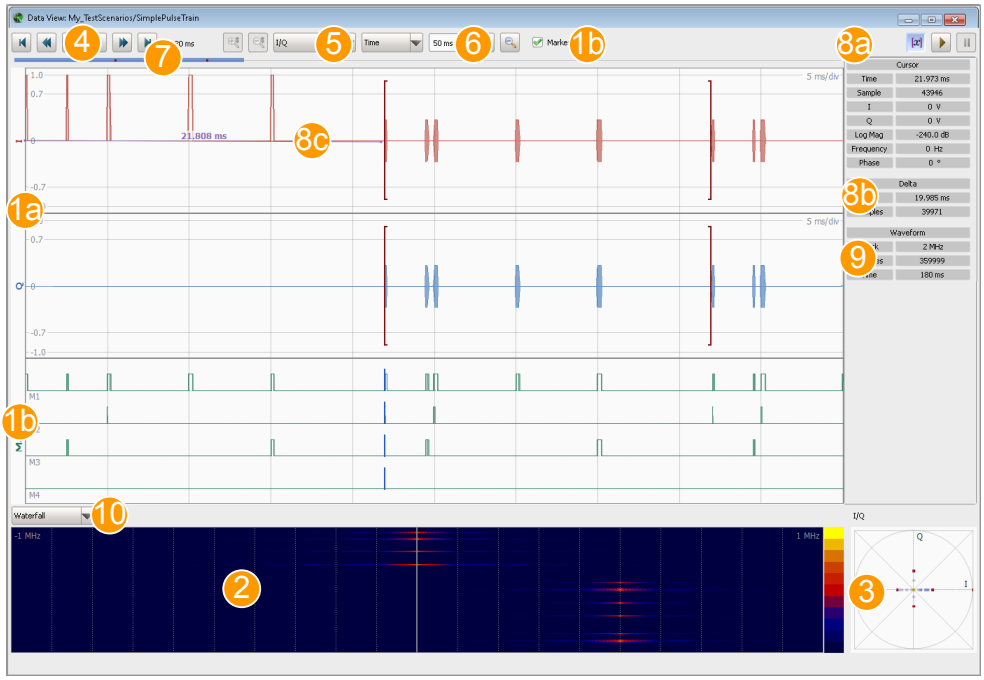


Fig. 18-1: Waveform/Data View: Understanding the displayed information

1a = Time domain view, "View mode = I/Q Envelope" (screen points represent multiple signal samples)
1b = Enabled marker traces (M1 = all, M2 = first, M3 = last), see Marker traces
2 = Waterfall diagram, see Display Mode
3 = I/Q constellation plot
4 = Navigation controls

5 = View mode, e.g. I/Q data, frequency, phase, magnitude
6 = Zoom in/out and signal part as number of samples or a time duration, see Duration/Samples
7 = Time line
8a = Delta cursors, play and stop; see Play/Stop
8b, 8c = Retrieved delta information
9 = File information: clock rate, number of samples and duration
10 = Display Mode selection

Navigation controls....241
View mode....241
Units....241
Zoom in/out....242
Marker traces....242
Duration/Samples....242
Time line....242
[x] Delta cursors, Delta markers....242
Play/Stop....242
Display Mode....242
I/Q constellation plot....243

Navigation controls
Steps forward or backward to the beginning or end of the waveform.

View mode
The window size and the screen resolution limit the number of screen points in the time domain display. This also limits the resolution at which level and time information is represented. Particularly when a large signal section is analyzed, one single screen point may contain the information from a large number of signal samples.

Depending on the zoom level, each screen point may represent multiple signal samples. If so, the area between the minimum and maximum value is filled with a semi-transparent trace color.

"I/Q" I and Q trace on a linear scale in the range between –1.0 and +1.0

"Magnitude dB/Watt/Volt"
Signal amplitude on a 100 dB/1000 mW/100 mV scale

"Phase" Phase trace on a linear scale in the range between –Pi and +Pi

"Frequency" Frequency trace on a linear scale with adjustable scale

"Average Power"
Average value when multiple samples are represented by one screen point.
See also chapter 18.2, "Waveform Reference Level Settings", on page 243.

Remote command:
`SCENario:VOLatile:VIEW:YMODe` on page 386

Units
Sets the units (time or samples) used on the x axis.

Remote command:
`SCENario:VOLatile:VIEW:XMODe` on page 386

Zoom in/out
Zooms in and out, i.e. changes the resolution on the x and y axis.

Remote command:
`SCENario:VOLatile:VIEW:ZOOM:POINt` on page 386
`SCENario:VOLatile:VIEW:ZOOM:RANGe` on page 387

Marker traces
In each of the view modes, enable "Marker > On" to display the up to four marker traces M1 to M4.

See also chapter 17, "Defining and Enabling Marker Signals", on page 236.

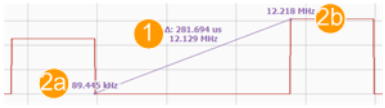
Duration/Samples
Sets the number of displayed signal samples, e.g. to display only a part of the entire waveform.

The "Waveform Viewer" displays a maximum of 100000 Samples. to visualize a longer part of a waveform, you can reduce the clock rate used by the calculation, see Clock, Duration.

Time line
- The length of the timeline symbolizes the play time of the entire waveform.
- The blue slider indicates the area of the waveform that is currently visible.
- The red dots indicate the time that corresponds to the mouse position.
- The current time is also displayed numerically above the timeline.
- A left mouse click adjusts the viewport center position to the selected time.
- Slide the blue slider to shift the visible waveform area.

[x] Delta cursors, Delta markers
To retrieve delta information, use one of the following:

"Temporary delta information"
Drag the mouse cursor on the diagram.
The delta information is displayed temporary above the line.



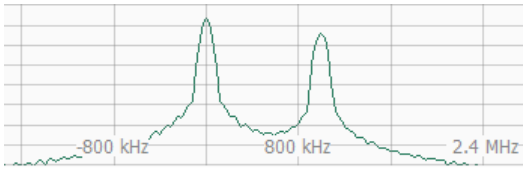
1 = Delta information: upper value is the Δx [in the selected "Units"], lower value is the Δ y [units depending on the selected "View mode"]
2a, 2b = Start and end y values [units depending on the selected "View mode"]

"[x] Delta cursors"
Select the "[x]" icon and Drag&Drop the cursors.
The delta cursors are indicated by brackets. The delta information is displayed in the "Delta" section of the dialog.

Play/Stop
Starts and stops the graphical waveform playback. The waveform section is shifted in time by one half of a display section every 100 ms.

Display Mode
Sets the display mode:

"Spectrum" The spectrum shows the FFT of the signal portion that is visible in the viewport. The FFT algorithm processes the waveform data in sections of 512 points.



"Waterfall diagram"
The waterfall shows the spectrum of a signal portion as a function of time. Each line in the waterfall diagram represents one FFT with 512 points.
The waterfall diagram uses a color scale to represent the dynamic range.

I/Q constellation plot
Depending on the number of samples visible in the viewport, displays the individual I/Q samples or turns to a density plot display.

18.2 Waveform Reference Level Settings

To access these settings:

1. Select "Repository Tree > Waveform".

2. Select "Waveform > Level".

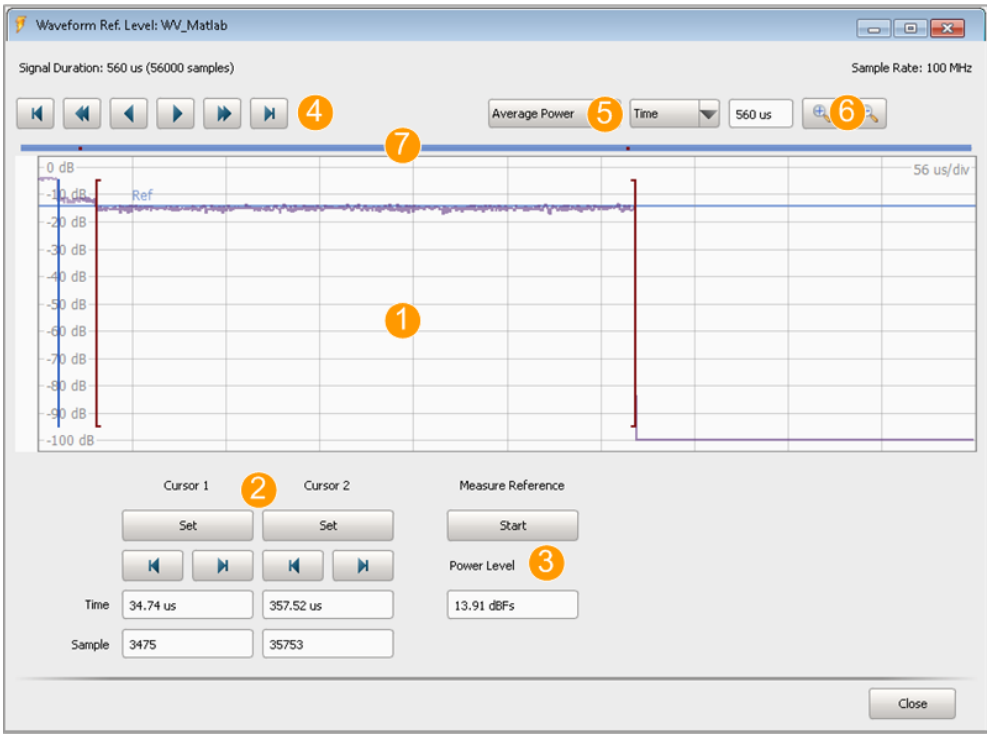


Fig. 18-2: Waveform Reference Level: Understanding the displayed information

1 = Time domain view, view mode = Average Power, see View mode
2 = Cursors
3 = Measured reference level
4 = Navigation controls
5 = View mode, e.g. I/Q data, power, etc.
6 = Zoom in/out and signal part as number of samples or a time duration, see Duration/Samples
7 = Time line

The "Waveform Reference Level" reads the entire waveform data and provides a view of the I/Q data versus time. The provided navigation controls, zooming functions, and view modes are similar to the functions provided in the "Waveform View" dialog, see figure 18-1.
The additional controls define the signal portion that is used for calculating of the reference signal level, see "To measure the reference signal level of a waveform" on page 246.

Cursors
Define the signal portion used for calculating of the reference signal level.

Measure Ref. Level
Calculates and displays the reference level of the waveform part, as defined with the current cursors.

18.3 How to Analyze the Content of Waveform Files and Files with I/Q Data

See:

- "To access the "Waveform/Data View" dialog" on page 245
- "To adjust the display and retrieve more information on the waveform" on page 245
- "To measure the reference signal level of a waveform" on page 246

To access the "Waveform/Data View" dialog

Perform one of the following:

1. Select "Repository Tree > Waveform" and:
   a) Select an imported waveform, e.g. "WV_Matlab".
   See "To import a custom waveform" on page 197.
   b) Select "Waveform > View".
2. Select "Repository Tree > Scenario" and select for example "SimplePulseTrain". We assume, that the scenario is configured, a signal generator is selected and a sequence is assigned to this scenario.

   See:

   - chapter 6.3, "How to Select and Create a Test Scenario", on page 63.
   - chapter 8.3, "How to Create Sequences and Use the Control Elements", on page 117.

   a) In the "Scenario" dialog, select "Start".
   The software calculates the waveform. Green LEDs indicate that the processing is completed.
   b) Select "Volatile > View".

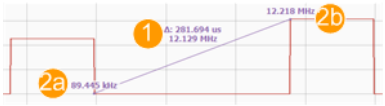
   See figure 18-1.
   The "Waveform/Data View" displays the I and Q data, the spectrum and the constellation diagram of the signal, as well as evaluated information from the waveform file, such as the clock rate, number of samples or the duration.

To adjust the display and retrieve more information on the waveform

1. Select the signal characteristic to be visualized, e.g. "View Mode > I/Q".
2. Set the cursor on a particular point to retrieve detailed information.

   A single cursor is indicated by a blue line. It sets the center point for all zoom operations.
3. Use the mouse wheel or the "Zoom In/Out" functions to display a part of the waveform or the whole generate waveform.
4. To retrieve delta information:

a) Select the "[x]" icon and Drag&Drop the cursors.
The delta cursors are indicated by brackets. The delta information is displayed in the "Delta" section of the dialog.
b) Drag the mouse cursor on the diagram.
The delta information is displayed temporary above the line.



1 = Delta information: upper value is the Δx [in the selected "Units"], lower value is the Δ y [units depending on the selected "View mode"]
2a, 2b = Start and end y values [units depending on the selected "View mode"]

5. Select "Waterfall" or "Spectrum".
6. If you observe a part of the waveform, select the play icon to playback the waveform and observe the signal changes over time.

To measure the reference signal level of a waveform

1. Select "Repository Tree > Waveform".
2. Select an imported waveform, e.g. "WV_Matlab".
See "To import a custom waveform" on page 197.
3. Select "Waveform > Level".

   The "Waveform Reference Level" dialog displays the I and Q data versus time, as the "Waveform View" does, but additionally calculates the reference signal level of the complete signal or a particular signal part.
   See figure 18-2.
4. Set the cursor on a particular point.

   A single cursor is indicated by a blue line. It sets the center point for all zoom operations.
5. Use the mouse wheel or the "Zoom In/Out" functions to display a part of the waveform or the whole waveform.
6. To measure the reference signal of a particular signal part:
   a) ● Set the cursor on the start point of the waveform interval to be measured or
      ● select "Cursor 1 > Time" and enter the beginning of the evaluation period.
   b) Select "Cursor 1 > Set".
   c) Perform the same for the second cursor.

   The two cursors are indicated by brackets. They indicate and define the beginning and end point for the reference level evaluation period.
7. Select "Measure Ref.".

   Evaluated are all samples in the defined period. The calculated reference signal level is displayed.

19 Playing the Generated Waveform Files

Complex pulse signals created with the R&S Pulse Sequencer can be played by:

- R&S SMW
- R&S SMBV

Depending on the installed option and the capabilities of the vector signal generator, files and waveforms are processed as an ARB file or in real time.

Waveform types, waveform handling and impact of the instrument's hardware on the calculated waveform

The R&S Pulse Sequencer determines the kind of waveform the signal generator instrument is able to handle. Supported are the following waveform types:

- **"Pure" ARB file**
  The R&S Pulse Sequencer calculates a *long waveform without sequencing*; calculation is performed offline and in advance
- **Multi Segment Waveform**
  The R&S Pulse Sequencer *pre-processes the sequence and splits it up into sequential* elements. These elements are then played using loops.

19.1 Waveform Generation Settings

In the general case, the software calculates a waveform, most suitable for the current signal generator; e.g. an ARB file or a multi segment waveform.

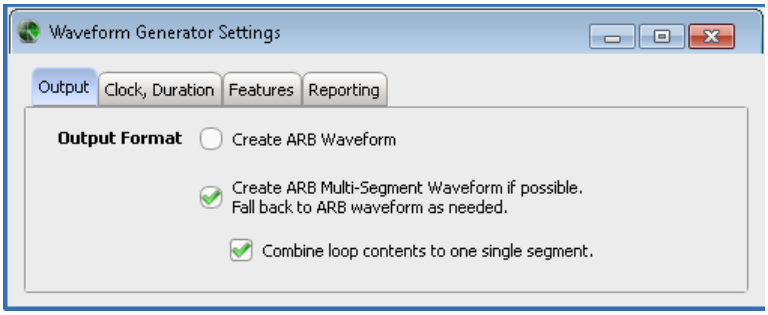
To access and change the settings influencing the waveform generation:

► In the "Scenario" dialog, select "Waveform Generation > Config".

Available are the following settings:

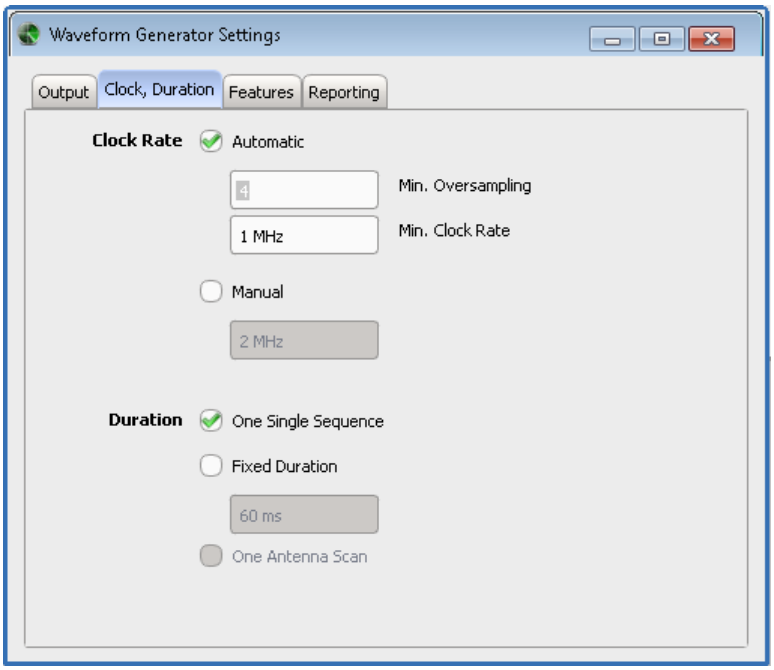
Output Format....247
Clock, Duration....248
Features....249
Reporting....250

Output Format
Defines the type of the generated waveform file, a single ARB file or an ARB multi segment waveform file.



Remote command:
`SCENario:OUTPut:FORMat` on page 370
`SCENario:OUTPut:LOOP:COMBine` on page 370

Clock, Duration



Provides clock related settings:

"Clock Rate" Retrieves the clock rate automatically form the generated waveform or sets a user-defined value.
The clock rate influences the size of the generated waveform. To reduce the size of long waveforms, change the clock rate.

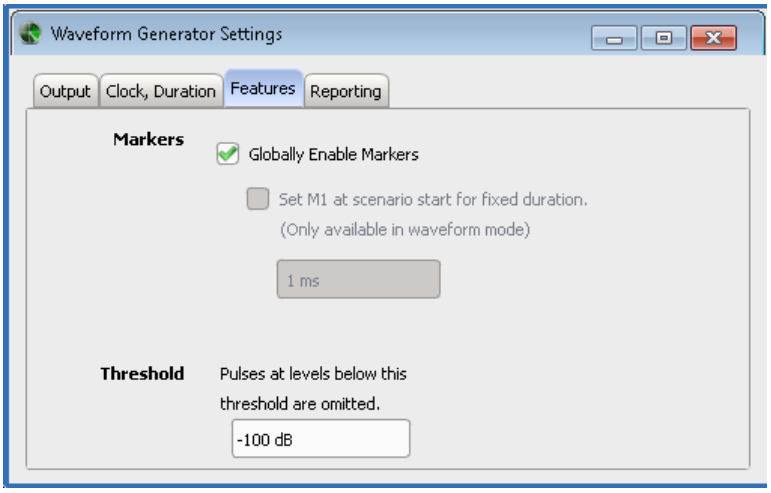
"Duration" Defines the waveform content.

The generated signal may:

- contain one single sequence; short sequences result in waveform files with small size
- last a specified duration; short sequences are repeated, longer once are truncated
- contain the signal describing one emitter antenna turn; short sequences are repeated, longer once are truncated

Remote command:
SCENario:OUTPut:CLOCk:MODE on page 368
SCENario:OUTPut:CLOCk:AUTO:BORDer on page 369
SCENario:OUTPut:CLOCk:AUTO:OVERsampling on page 369
SCENario:OUTPut:CLOCk:USER on page 369
SCENario:OUTPut:DURation:MODE on page 369
SCENario:OUTPut:DURation:TIME on page 370

Features



Sets:

"Markers" Enables that the configured markers are considered in the generated waveform file.
See "Global markers" on page 238.
You can also enable an additional marker, that is held high from the scenario start until a selected duration. This marker signal is output as marker 1. If other marker signals are defined, all markers are summed.

"Threshold" Enters the threshold. Pulses at levels below this threshold are omitted.

Remote command:
SCENario:OUTPut:MARKer:ENABle on page 325
SCENario:OUTPut:MARKer:SCENario:ENABle on page 325
SCENario:OUTPut:MARKer:SCENario:DURation on page 325
SCENario:OUTPut:THReshold on page 372

Reporting
Enables and configures the report files generation, see chapter 20, "Creating Reports and Documenting Measurement Results", on page 260.

19.2 Signal to Generator Mapping Settings

In a "Scenario Type > Localized Emitters", you have to select the signal of which emitter/interferer is currently transmitted to and processed by the connected instrument.

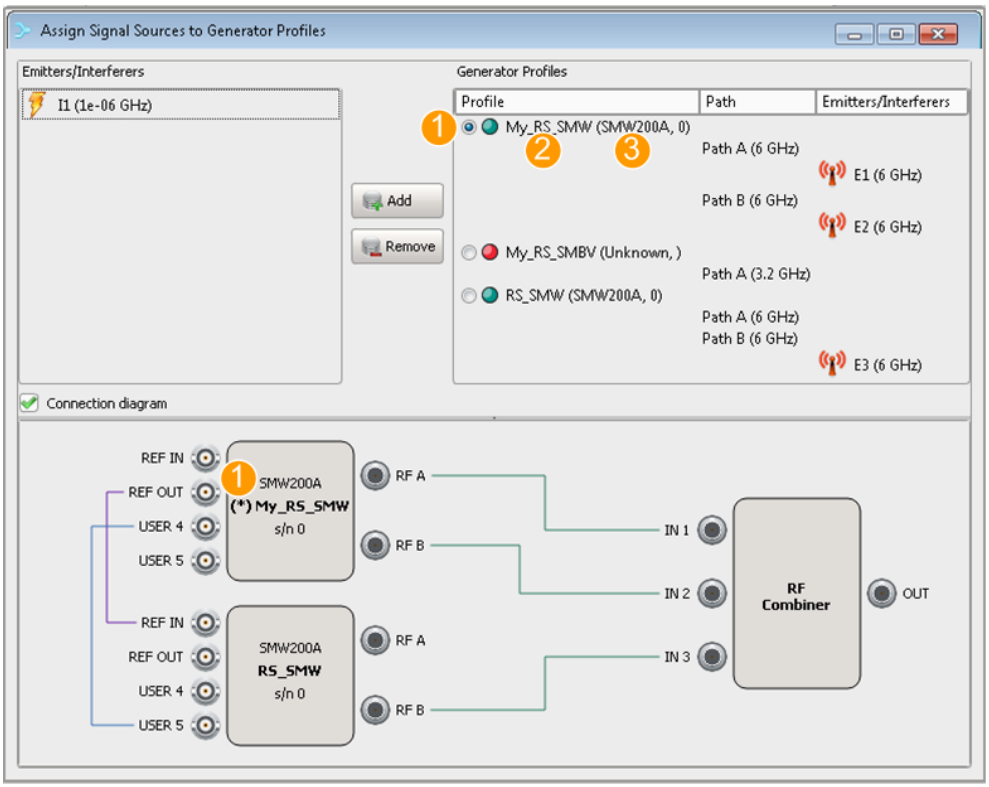
You can assign more then one signal to each of the instrument paths.

See for example:

- chapter 13.2, "How to Create Scenarios with Emitters, Interferers and a Receiver", on page 209
- "To map the generated signals to the signal generator" on page 214

To access these settings:

► In the 2D map of a "Scenario Type = Localized Emitters", select "Assign Emitters".



1 = Master instrument
2 = Generator profile (see chapter 14.2, "Generator Profile Settings", on page 219)
3 = Physical instrument and serial number

Emitters/Interferers....251
Generator Profiles....251
Add, Remove....251
Connection diagram....251

Emitters/Interferers
Lists the alias names of all currently configured emitters and interferers, that are still not assigned to a generator.

Remote command:
`ASSignment:EMITters:LIST?` on page 302
`ASSignment:EMITters:SELect` on page 280

Generator Profiles
Lists the available generator profiles with their paths, supported maximum RF per path and assigned emitters/interferers.

The color LED are status indication of the mapping status:
- **Green** - a physical instrument is mapped to the profile.
  The signal generator is indicated with its name and serial number (see also the List of connected instruments in the "Instruments Config" dialog)
  Status "Not mapped" indicates that the generated waveforms are stored as locally ("Local ARB File > Target > File").
- **Red** - no signal generator is mapped

Remote command:
`ASSignment:GENerator:LIST?` on page 302
`ASSignment:GENerator:SELect` on page 280
`ASSignment:GENerator:CAPabilities?` on page 302
`ASSignment:GENerator:PATH:LIST?` on page 302
`ASSignment:GENerator:PATH:SELect` on page 280
`ASSignment:GENerator:PATH:EMITter:LIST?` on page 303

Add, Remove
Adds (or removes) a selected emitter/interferer to a selected generator's path.

Alternatively, use the Drag&Drop method.

Remote command:
`ASSignment:GENerator:PATH:EMITter:ADD` on page 279
`ASSignment:GENerator:PATH:EMITter:SELect` on page 280
`ASSignment:GENerator:PATH:EMITter:DELete` on page 281

Connection diagram
Visualizes the required physical connections between the signal generator(s) and the DUT.

In test setups with more than one signal generators, the **(*)** sign indicates the signal generator that acts as a master device.

Provide all other instruments with the reference frequency and the trigger signal of the master signal generator.

See also chapter 19.4.2, "How to Play the Waveforms of Emitter-Based 2D Scenarios", on page 255.

19.3 Signal Generator Remote Control Settings

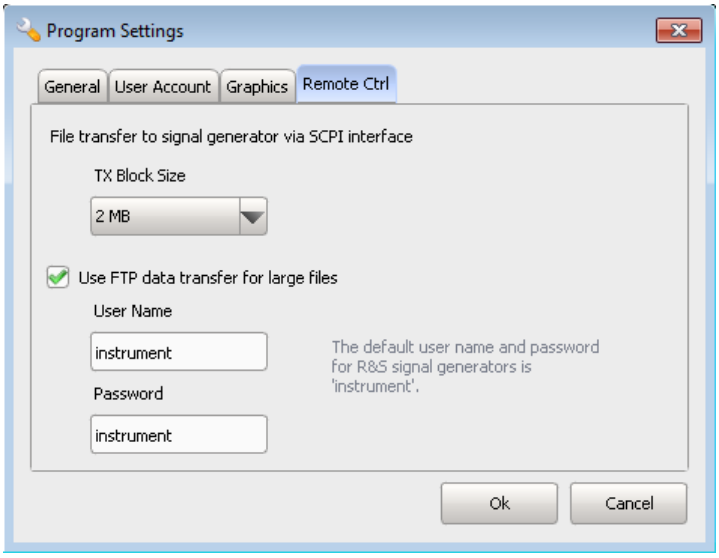
Waveform files generated with the software are automatically uploaded to the connected signal generator in one of the following alternative ways:

- via the SCPI interface
- via FTP transfer
  If enabled, this second option is automatically applied when the calculated files are large.

Both methods require that the signal generator and the PC on that the software is installed are connected to the same LAN.

To access the required settings:

1. In the "Menu bar", select "Configure > Settings > Remote Control".



2. If required, change the size of the transport blocks transmitted via SCPI interface.
3. To make use of the FTP data transfer:
   a) In the signal generator, enable the FTP interface.
   You find the required settings in the "Setup > Security" dialog.
   For more information, see the signal generator's embedded help or refer to its user manual (see chapter A.6, "References", on page 412).
   b) In the R&S Pulse Sequencer, enable a FTP data transfer to speed up the data upload to the connected signal generator.
   The FTP transmission requires "User Name" and "Password". The default "User Name" and "Password" is *instrument*.

Note: The R&S Pulse Sequencer will attempt to connect to *any* connected signal generator with the *same credentials* (i.e. "User Name" and "Password").

Depending on the file size, the software will use either the SCPI interface or the FTP data transfer.

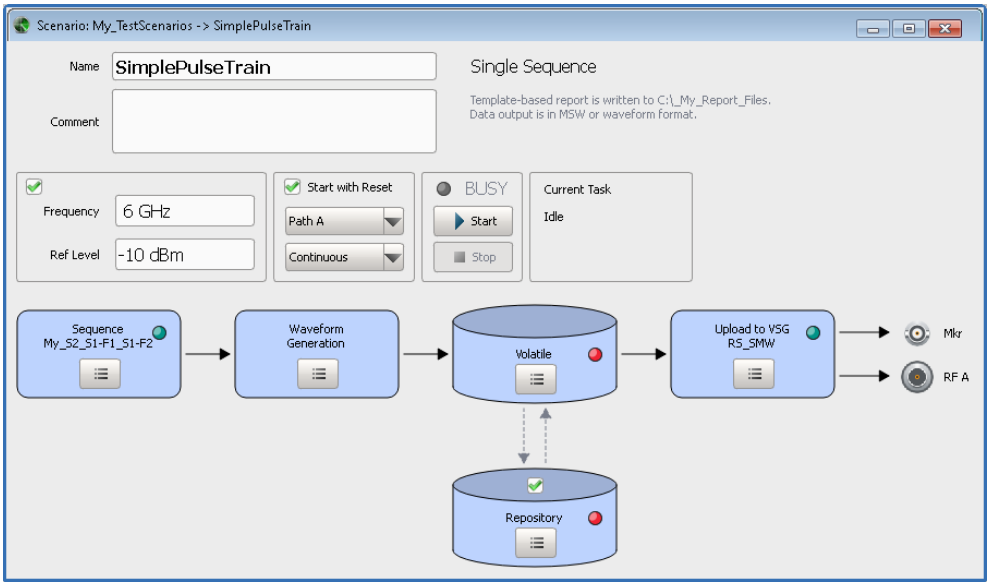
19.4 Playing Waveforms with the Signal Generator

The following examples explain the principle and use R&S SMW to play the waveforms.

19.4.1 How to Create, Transfer and Play the Waveforms of Simple Scenarios

This examples uses the test setup as shown on figure 4-1

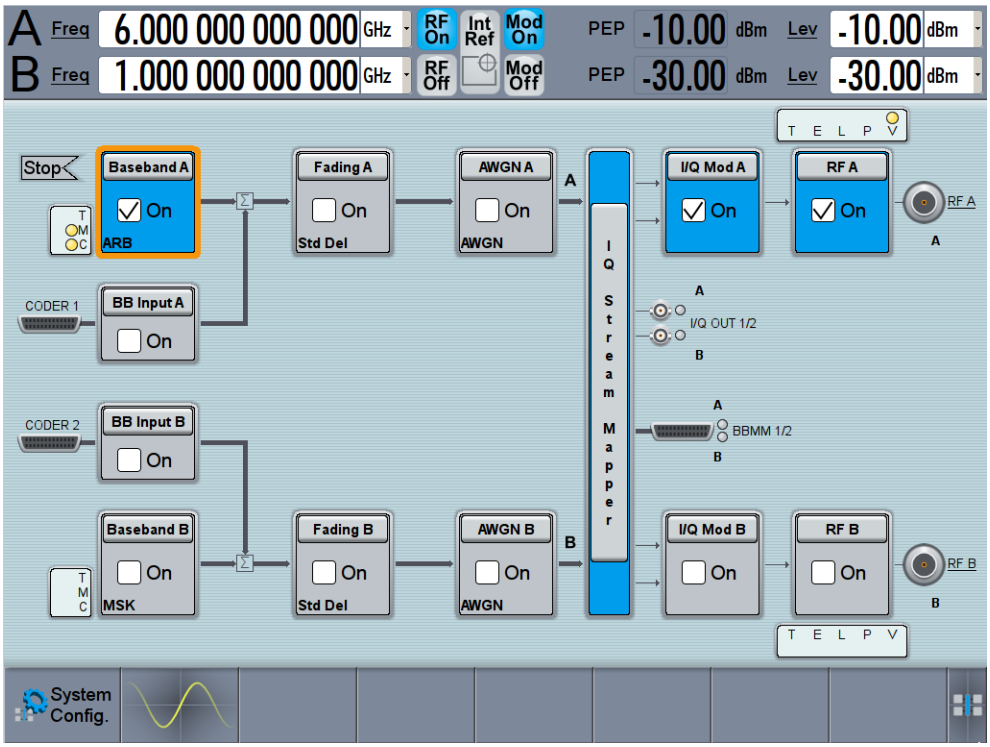
1. Open a suitable scenario, e.g. "SimplePulseTrain"
   See "To create a single pulse train scenario" on page 64.
2. Configure the settings as follows:



   See:
   - chapter 8.3, "How to Create Sequences and Use the Control Elements", on page 117
   - chapter 14.3, "How to Create Generator Profiles and Configure the Connected Instruments", on page 221
3. Use the default "Waveform Generator" settings.
   See also chapter 19.1, "Waveform Generation Settings", on page 247.

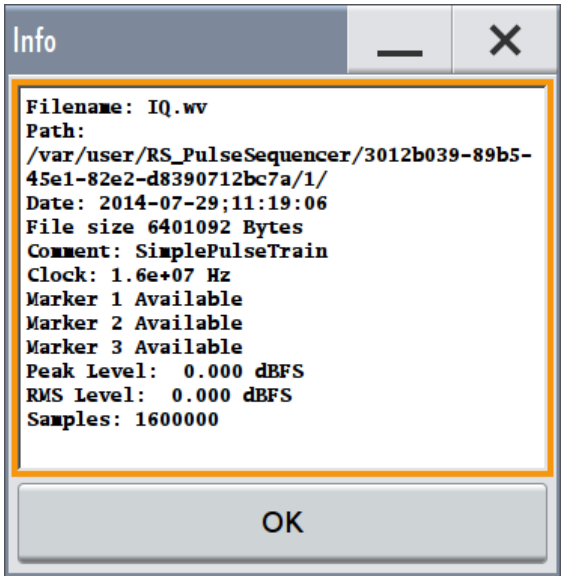
   All prerequisites are fulfilled. The "Start" button is enabled.
4. Select "Start".
   - The software calculates the waveform and stores it in the "Volatile" memory
   - Progress information indicates the calculation progress
   - The waveform is automatically transferred to the instrument and loaded in the "ARB"
   - The output frequency and level of the signal generator are set accordingly

- Signal generation is started and the signal is output at the RF A connector



In the signal generator, perform further configurations, e.g.:

- Select "Baseband > ARB > General > Waveform Info" to confirm that the waveform generated with the R&S Pulse Sequencer.
  You retrieve also information on the waveform file name, used clock rate, storage location, etc.



- The generated signal contains marker signals.
  Use the default routing or change the mapping of the marker signals to the output connectors.

For details, see description R&S®SMW200A User Manual.

In the R&S Pulse Sequencer:

- Select "Volatile > View" to visualize the generated waveform.
  The display should resemble the one on figure 8-6.

Playing the waveforms of 2D scenarios

If your current scenario type is "Localized Emitters", you have to additionally:

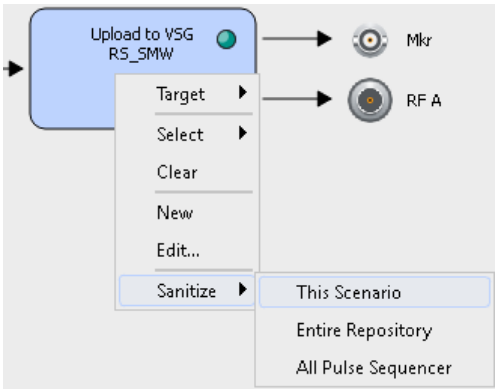
- select the signal of which emitter/interferer is transmitted to the connected signal generator
- in a test setup with two or more signal generators:
  – select "Connection diagram > On"
  – define the master device
  – connect the instruments as illustrated on the diagram

For detailed description, see chapter 19.4.2, "How to Play the Waveforms of Emitter-Based 2D Scenarios", on page 255.

To delete wavefroms form the hard disk of the signal generator

If the waveforms uploaded on the vector signal generator are no longer required, they can be deleted automatically form the R&S Pulse Sequencer:

1. Open the "Scenario" dialog.
2. In the context menu of the "Upload to VSG" block, select "Sanitize" and select e.g. "This Scenario".



The waveforms belonging to the current scenario are deleted from the hard disk of the signal generator.

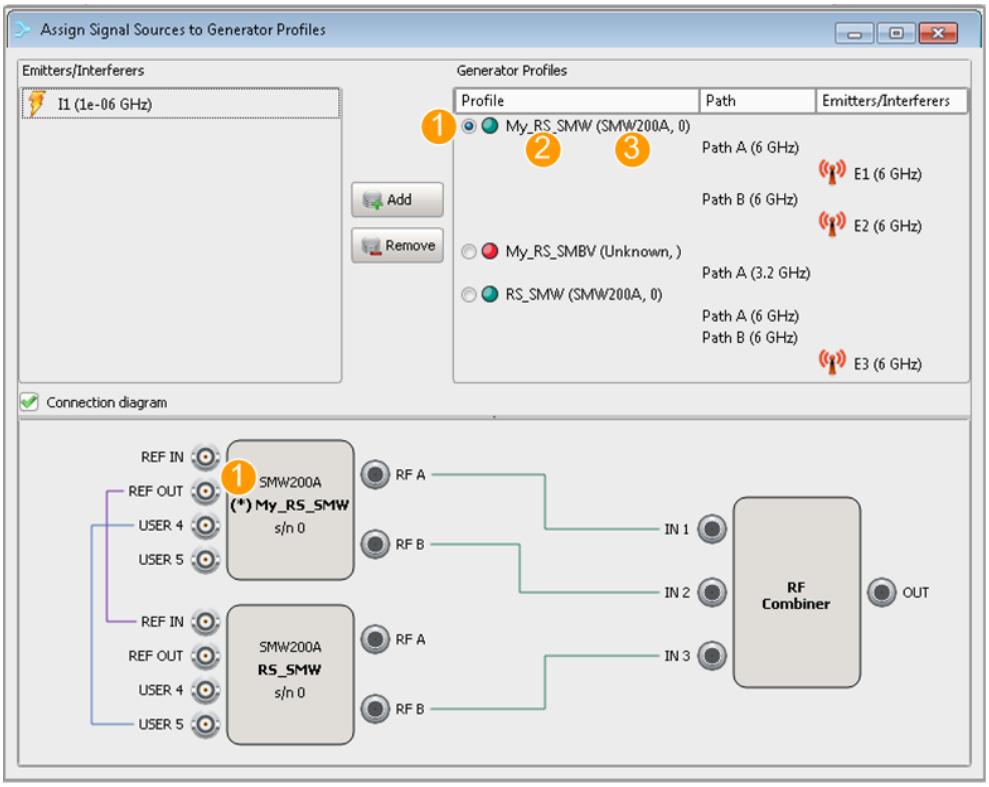
19.4.2 How to Play the Waveforms of Emitter-Based 2D Scenarios

To create, load and play the waveforms of a 2D scenario automatically

1. Open a test scenario with "Scenario Type > Localized Emitters", e.g. "LocalizedEmitters".

(see chapter 13.2, "How to Create Scenarios with Emitters, Interferers and a Receiver", on page 209)

2. Select "Upload to VSG > Target > Generator".
3. Select "Upload to VSG > Assign Emitters"
4. Drag&Drop an Emitter/Interferer to assign its signal to a generator profile.



1 = Master instrument
2 = Generator profile (see chapter 14.2, "Generator Profile Settings", on page 219)
3 = Physical instrument and serial number

5. Enable "Connection diagram > On".

   The diagram visualizes the required connections.
   The **(*)** sign indicates the signal generator that acts as a master device.

6. In "Generator Profile" section, change the master device according to your particular setup.
7. Provide all other instruments with the reference frequency and the trigger signal of the master signal generator.

   The R&S Pulse Sequencer sends the required SCPI commands to the instruments. All required signal generator settings are configured automatically.
   In this example for instance, the direction and the mapped signal to USER 4 and USER 5 connectors of both signal generators are configured automatically.

8. In the scenario dialog, select "Start".

The software calculates the waveforms. Progress information indicates the calculation progress.
The waveforms are stored in the volatile memory, transferred to the instruments and loaded in the "ARB"s.
The output frequency and level of the signal generators are set accordingly.
Signal generation is started synchronous with a *fixed delay of 1 ms*; the signal is output at the RF connectors.

9. In the block diagram of the current scenario, select "Volatile > View > E1" to visualize the waveform of the emitter "E1".

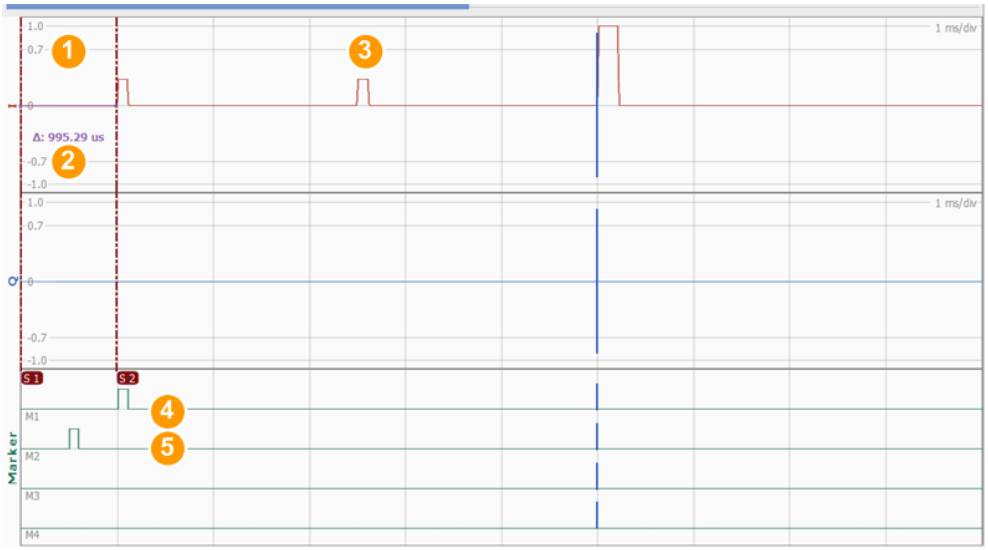


Fig. 19-1: Data View: Example of a waveform in a test setup with multiple instruments

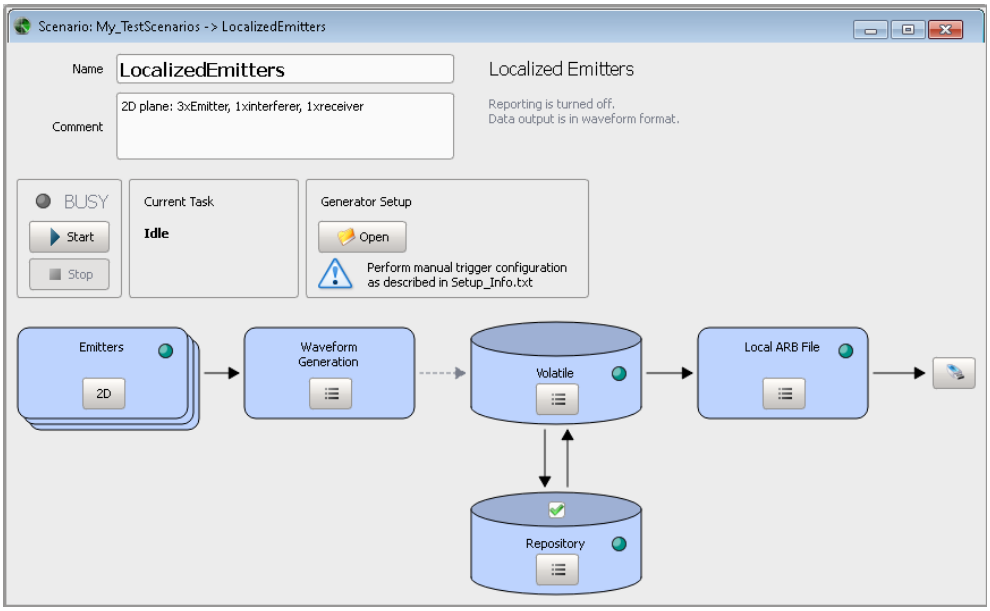
1 = Automatically appended blank segment
2 = Blank segment with fixed duration of 1 ms
3 = Second segment
4 = Marker M1: the sequence start marker which appears at the first pulse of the sequence; M1 can be used to trigger e.g. an analyzer to the start of the scenario
5 = Marker M2: automatically generated marker at the middle of the blank segment; M2 is used for synchronization purposes

The "Data View" shows that a blank segment of 1 ms is appended to the waveform. In the middle of this segment, i.e. after 500 us, the automatically generated marker signal M2 appears (see figure 19-1). This marker M2 is used to synchronize test setups with multiple instruments.
The master instrument, the R&S SMW in this example, is triggered by the R&S Pulse Sequencer. Once the waveform is loaded into the ARB of the master instrument, the blank segment is played first. The marker signal M2 triggers the ARB of the slave instrument, this is the second R&S SMW. The duration of the blank segment and the time interval of 500 us are selected to be longer than the processing time of the slave instrument. An internal procedure guarantees that the period from generating the marker signal M2 in the master instrument until start of the ARB in the slave instrument is exactly 500 us.
Thus, the ARBs of all signal generators in the setup start synchronously.

To create, transfer and play the waveforms of 2D scenario manually

During the initial tests it is useful to create test waveforms and store them on a local or network drive. These waveforms can be latter transferred to and played with a signal generator.

1. Open a test scenario with "Scenario Type > Localized Emitters", e.g. "LocalizedEmitters".
   (see chapter 13.2, "How to Create Scenarios with Emitters, Interferers and a Receiver", on page 209)
2. Select "Upload to VSG > Target > File".
3. Select "Local ARB File > Set Path" and select the storage location for the generated waveform.
4. Select "Repository > On".
5. Select "Start".



   The software calculates the waveform and stores it in the "Volatile" memory. Progress information indicates the calculation progress.
   The waveforms are created, automatically named and stored in the selected directory.
6. If your user rights are sufficient, you can store the generated file also in the repository.
   Select "Volatile > Release to Repository".
7. Select "Generator Setup > Open".

   In this example, the directory contains the `Setup_Info.txt` file and three waveforms (`My_RS_SMW_PathA_1_E1.wv`, `My_RS_SMW_PathB_3_E2.wv` and `RS_SMW_PathA_4_E3.wv`), one for each mapped emitter.

8. Open the `Setup_Info.txt` file.

```
Repository:     My_TestScenarios
Scenario:       LocalizedEmitters
Creator:        R&S
Date:           Mi 21. Jan 11:24:25 2015

Setup table

 Generator   |  Path  |      Waveform File      | Frequency |    Level    | Trigger offset |
-------------------------------------------------------------------------------------------
My_RS_SMW    | Path A | My_RS_SMW_PathA_1_E1.wv |   6 GHz   |   -1.1 dBm  |   11.082 us    |
My_RS_SMW(*) | Path B | My_RS_SMW_PathB_3_E2.wv |   6 GHz   |   3.54 dBm  |       0  s     |
RS_SMW       | Path A | RS_SMW_PathA_4_E3.wv    |   6 GHz   |  46.64 mdBm |    7.769 us    |
-------------------------------------------------------------------------------------------

Master Emitter/Interferer in table (marked with asterisk) should trigger all other Emitters/Interferers.
Connect the trigger signals according to the block diagram and configure the trigger modes of the individual basebands manually
Ensure, that the reference clock of the signal generator with the master Emitter/Interferer is distributed to all other generators
```

9. Follow the instructions listed in the file.
10. To display the test setup and the connection diagram, select "Emitters > 2D > Assign Emitters" and enable "Connection diagram > On".

20 Creating Reports and Documenting Measurement Results

In R&S Pulse Sequencer, you can document your measurement results and software configuration in the fallowing ways:

- create and export a repository archive, so that you can repeat measurements under the same conditions
- create automatically scaled hardcopy of current screens or dialogs
- print antenna diagrams together with additional information
- create and store reports in text files or in Excel files with several spread sheets.

This section focuses on the built-in report generation function. For information on the other functions, see:

- "To create and export a repository archive" on page 50
- "To set the size and color scheme of the hardcopies" on page 37
- chapter 10, "Defining Antenna Patterns and Antenna Scans", on page 154

Report generation

The R&S Pulse Sequencer provides a built-in report generation function so that all parameters used during the waveform generations are automatically stored into a printable file. The directory the generated reports are stored in is configurable.

Report files can be foramted based on:

- **Template**
  The reporting uses a predefined but editable template. Generated is an ASCII text file where data is formatted in columns, including header description. The reporting file uses the predefined file extension `*.pwd`.

```
# Pulse Sequencer PDW Report
#
Date:       2014-11-28T10:41:25
Repository: My_TestScenarios
Scenario:   SimplePulseTrain

          TOA |        RF |        PW |     PA | MF |  MOP |     BW |
           ns |       GHz |        us |    dBm |    |      |    kHz |
-----------------------------------------------------------------------
            0 |  3.000000 |   120.000 |    0.0 |  0 | NONE |      0 |
      2500000 |  3.000000 |   120.000 |    0.0 |  0 | NONE |      0 |
      5000000 |  3.000000 |   220.000 |    0.0 |  0 | NONE |      0 |
     10000000 |  3.000000 |   220.000 |    0.0 |  0 | NONE |      0 |
     15000000 |  3.000000 |   220.000 |    0.0 |  0 | NONE |      0 |
     22000000 |  2.999500 |   120.000 |  -10.0 |  0 | NONE |      0 |
     24500000 |  2.999500 |   120.000 |  -10.0 |  0 | NONE |      0 |
     25000000 |  2.999500 |   220.000 |  -10.0 |  0 | NONE |      0 |
     30000000 |  2.999500 |   220.000 |  -10.0 |  0 | NONE |      0 |
```

Fig. 20-1: Example of a text report file ("Type > Template")

- **User-defined plugin**

This reporting creates reports according to custom templates, like DFS, DIV8 PDW, etc.
You may for example create a Microsoft Excel plugin for example, that retrieves the pulse parameters and it fills them in a spread sheet.

For detailed description of the file formats, the template and the plugin, see:

- chapter A.1.1, "File Format of the Reporting Template", on page 390
- chapter A.3, "Plugin Programming API", on page 400
- chapter 16, "Defining Complex Modulation Schemes and IPM Profiles", on page 232.

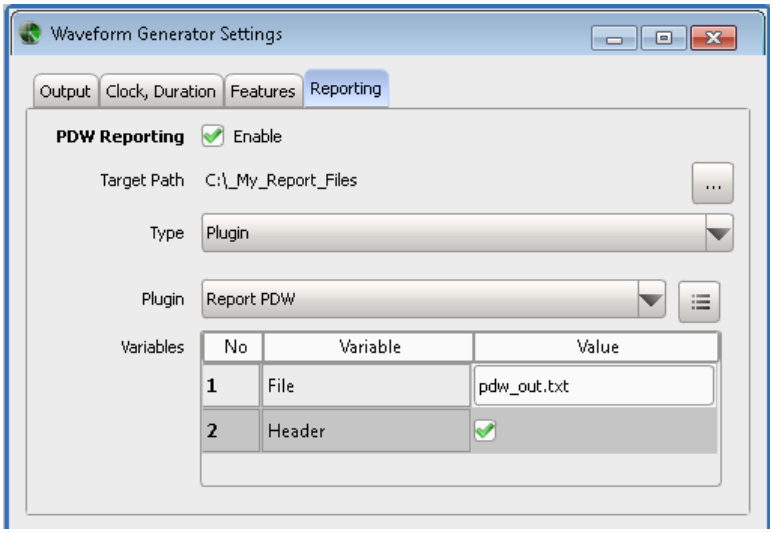
Status information on whether report logging is enabled or not is displayed in the "Scenario" dialog.

See also chapter 20.2, "How to Create Test Reports", on page 263.

20.1 Reporting Settings

To access and change the settings influencing the report generation:

1. In the "Scenario" dialog, select "Waveform Generation > Config".
2. Select "Reporting".



Available are the following settings:

Reporting Enable..... 262
Target Path, Set Path..... 262
Type..... 262
Edit Template..... 262
Plugin and Plugin Variables..... 263

Reporting Enable
Enables generation of reports.

Remote command:
SCENario:PDW:ENABle on page 329

Target Path, Set Path
Indicates current storage location.

Per default, report logs are stored in the user directory of the current user, that is the %HOMEPATH% directory.

To change the storage location, select the "Set Path" icon.

See also "To set the storage location for the generated report files" on page 264.

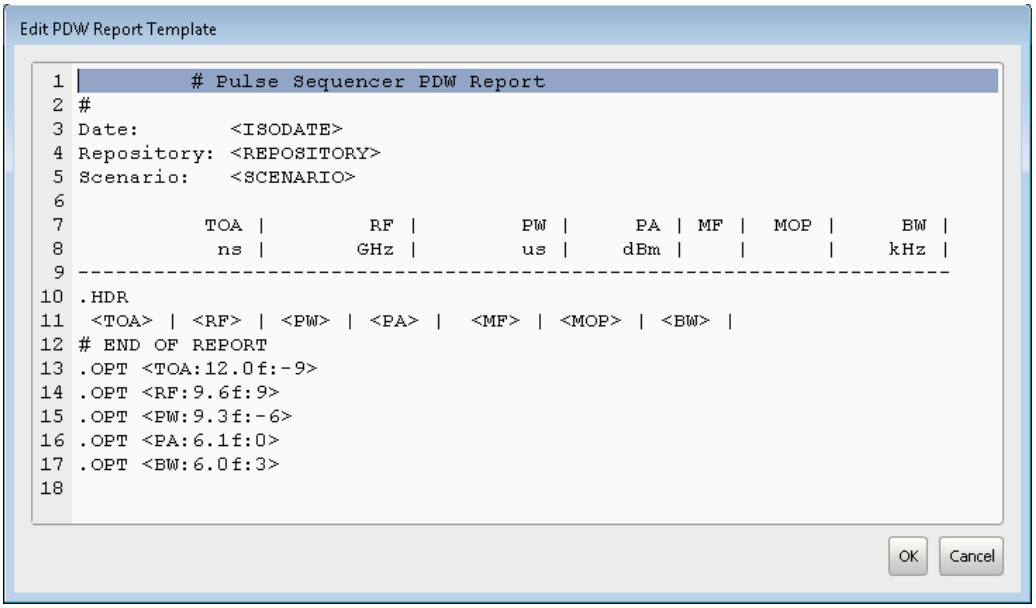
Remote command:
SCENario:PDW:PATH on page 329

Type
Sets the template used be the reporting function.

"Default"	Loads the predefined Template. The report is a plain text file. The format is simple, column-based and cannot be modified. Further settings are not required.
"Template"	Enables the predefined template for editing, see "Edit Template" on page 262.
"Plugin"	Enables loading according to custom templates, like DFS, DIV8 PDW, etc. Custom templates are used in the same way as the predefined template, see "Plugin and Plugin Variables" on page 263.

Remote command:
SCENario:PDW:TYPE on page 330

Edit Template
Opens the predefined template in edit mode.



See chapter A.1.1, "File Format of the Reporting Template", on page 390 for description of the file format.

Remote command:
`SCENario:PDW:TEMPlate` on page 330

Plugin and Plugin Variables
Selects and loads a reporting template form a `*.dll` file. This template must exist in the "Plugin" library.

Report plugins may register a set of variables that can be used to further control the plugin's algorithms.

See:
- chapter A.3, "Plugin Programming API", on page 400
- chapter 16, "Defining Complex Modulation Schemes and IPM Profiles", on page 232.

Remote command:
`SCENario:PDW:PLUGin:NAME` on page 330
`SCENario:PDW:PLUGin:VARiable:CATalog?` on page 329
`SCENario:PDW:PLUGin:VARiable:SELect` on page 329
`SCENario:PDW:PLUGin:VARiable:VALue` on page 329

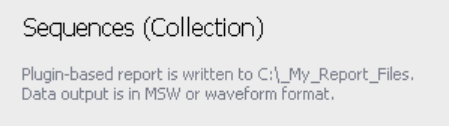
20.2 How to Create Test Reports

This section shows how to:

- "To enable or disable report file generation" on page 264
- "To set the storage location for the generated report files" on page 264
- "To change the report type" on page 264

To enable or disable report file generation

The "Scenario" dialog provides information on the report logging status. The related information is displayed to the right of the "Comment" field.



To enable or disable the report file generation, proceed as following:

1. In the "Scenario" dialog, select "Waveform Generation > Config".
2. In the "Waveform Generation Settings" dialog, select "Reporting > Reporting > Enable/Disable".

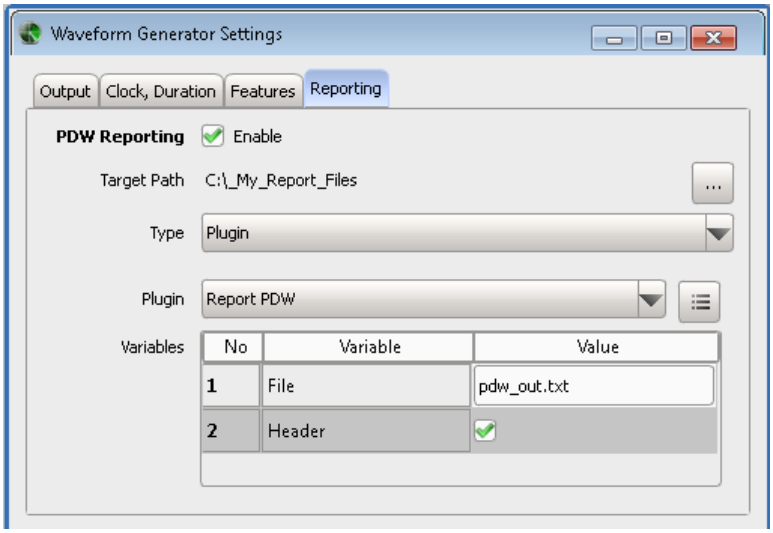
To set the storage location for the generated report files

Per default, report files are stored in the user home directory of the current user, that is the `%HOMEPATH%` directory.

1. Open the "Scenario" dialog to retrieve information on the current storage location.

   Information is displayed to the right of the "Comment" field.
2. Select "Waveform Generation > Config > Reporting".

   The parameter "Target Path" indicates the current storage location.



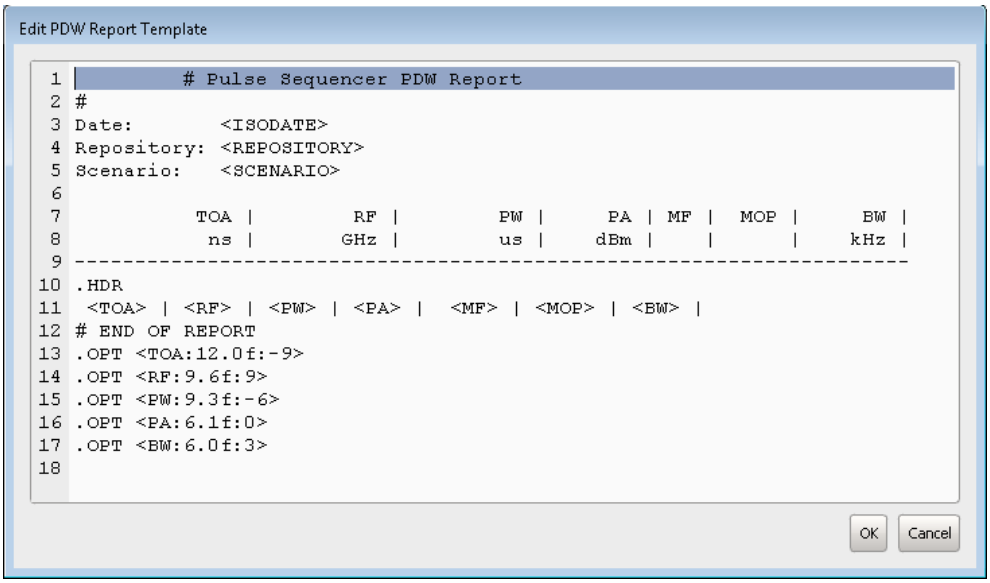
3. To change this location, select the "Set Path" icon, navigate to the directory and confirm with ok.

To change the report type

You may prefer to store the test reports in text form, if for example Microsoft Excel is not installed on your PC.

To change the type of the generated reports proceed as following:

1. In the "Scenario" dialog, select "Waveform Generation > Config > Report".
2. Select "Type > Template".
3. Select "Edit" to display the current template.



If required, you can also change the used templates or create and load your plugin.

See:

- chapter A.1.1, "File Format of the Reporting Template", on page 390
- chapter A.3, "Plugin Programming API", on page 400
- chapter 16, "Defining Complex Modulation Schemes and IPM Profiles", on page 232.

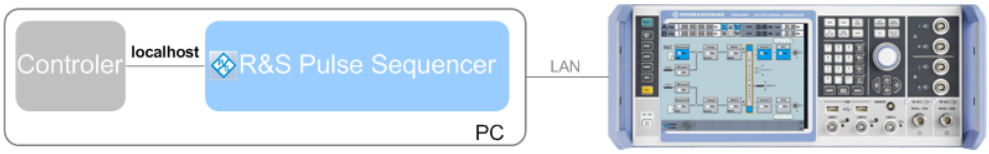
21 Automation of R&S Pulse Sequencer

In addition to working with the R&S Pulse Sequencer software interactively, it is also possible to operate and control it from a remote PC. Remote control operation allows you to automate the configuration process and is useful when a higher configuration speed is required.

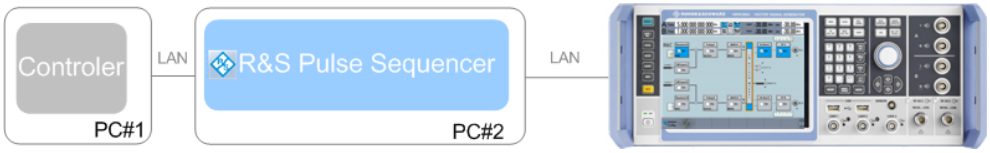
Remote control is an operation by which the software is operated remotely from a so called controller. A controller can be any PC on which some required software is installed and configured. R&S Pulse Sequencer might be but not necessarily is installed on the same PC. In case R&S Pulse Sequencer is not physically located on the controller's PC, a LAN connection between both PCs is required.

The figures below depict the following two possible cases:

- The controller and R&S Pulse Sequencer are on the same PC.



- In some rare cases, the controller and the software are installed on two separate PCs.



In remote control operation, the software itself is operated by means of remote control commands. For description of specific SCPI commands, refer to chapter 22, "Remote-Control Commands", on page 269.

We assume that the controller and the signal generator have already been set up for remote operation in a network as described in the documentation of the instrument. A knowledge about the remote control operation and the SCPI command syntax are assumed.

Supported interfaces and protocols

R&S Pulse Sequencer support remote control over the LAN interface and with socket protocol only. The socket controller is sufficient; a VISA (Virtual Instrument Software Architecture) library is not required.

Socket communication requires the specification of the port (commonly referred to as port number); the registered port is 5025.

Socket communication

Socket communication is a simple network communication and is also referred to as "Raw Ethernet communication". It is available by default on all operating systems.

The simplest way to establish socket communication is to use the built-in Telnet program. The Telnet program is part of every operating system and supports a communication with the software on a command-by-command basis. For more convenience and to enable automation by means of programs, user-defined sockets can be programmed.

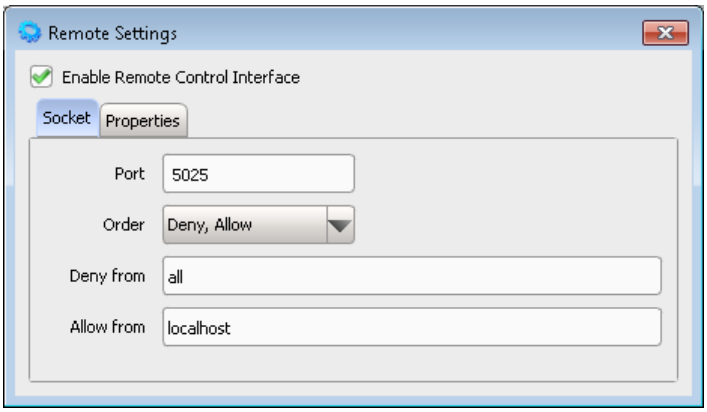
21.1 How to Configure and Enable Remote Control of R&S Pulse Sequencer

See:

- "To access the required settings" on page 267
- "To use the console to test SCPI commands" on page 268
- "To find the SCPI command corresponding to a parameter on the user interface" on page 268

To access the required settings

1. In the menu bar, select "Configure > Remote Control".
2. Select "Enable Remote Control Interface".
3. Do not change the default socket port.

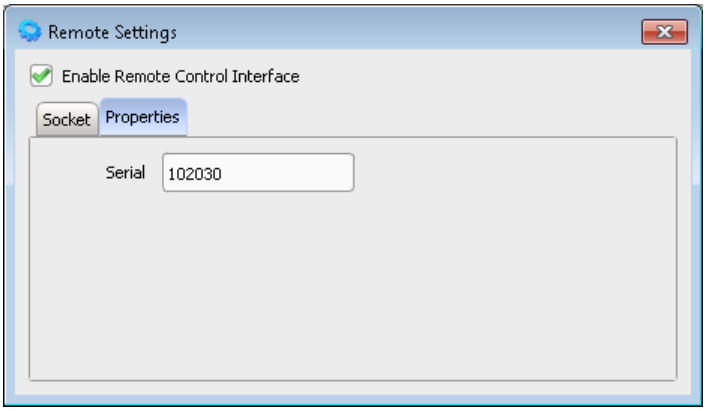


To identify the R&S Pulse Sequencer by its serial number

If you remotely control several R&S Pulse Sequencer from the same controller, it is useful to use serial numbers to distinguish between the different installations.

To set the serial number of the software:

1. In the menu bar, select "Configure > Remote Control".
2. Open the "Properties" tab.
3. In the "Serial" field, enter an unique six digit number.

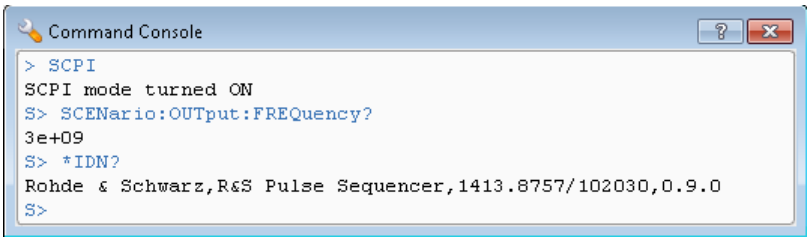


To query the serial number, use the command `*IDN?`.
See also "To use the console to test SCPI commands" on page 268.

To use the console to test SCPI commands

R&S Pulse Sequencer provides a built-in console window that allows you to test SCPI commands or commands sequences.

1. To access this window, in the menu bar, select "Window > Console"
2. In the "Command Console" window, type `SCPI` to start the remote control mode.
3. Enter any SCPI command, see "To use the console to test SCPI commands" on page 268.



To find the SCPI command corresponding to a parameter on the user interface

Perform one of the following:

1. Right mouse click to open the context menu of the parameter.
2. Press F1 to open the Help window. You find a link to the description of the SCPI command right after the parameter description.

   For a concise description of the available SCPI commands, see chapter 22, "Remote-Control Commands", on page 269.

22 Remote-Control Commands

The following commands are required to perform signal generation with the R&S Pulse Sequencer option in a remote environment. We assume that the R&S Pulse Sequencer has already been set up for remote operation in a network as described in the R&S Pulse Sequencer documentation. A knowledge about the remote control operation and the SCPI command syntax are assumed.

- Conventions used in SCPI Command Descriptions....269
- Programming Examples....270
- Common Commands....270
- Commands with Similar Syntax....271
- Antenna Pattern Commands....283
- Antenna Scan Commands....291
- Data Source Commands....299
- Emitter to Generator Mapping Commands....301
- Emitter Commands....304
- Generator Profiles and Instruments Commands....307
- Inter Pulse Modulation Commands....310
- Marker Commands....322
- Plugin and Reporting Commands....326
- Pulse Commands....331
- Repository Commands....350
- Scenario Commands....357
- Sequence Commands....372
- Status Commands....381
- System Commands....382
- Waveform Commands....384
- Waveform Viewer Commands....385

22.1 Conventions used in SCPI Command Descriptions

Note the following conventions used in the remote command descriptions:

- **Command usage**
  If not specified otherwise, commands can be used both for setting and for querying parameters.
  If a command can be used for setting or querying only, or if it initiates an event, the usage is stated explicitely.
- **Parameter usage**
  If not specified otherwise, a parameter can be used to set a value and it is the result of a query.
  Parameters required only for setting are indicated as **Setting parameters**.
  Parameters required only to refine a query are indicated as **Query parameters**.
  Parameters that are only returned as the result of a query are indicated as **Return values**.
- **Conformity**

Commands that are taken from the SCPI standard are indicated as **SCPI confirmed**. All commands used by the R&S Pulse Sequencer follow the SCPI syntax rules.

- **Asynchronous commands**
  A command which does not automatically finish executing before the next command starts executing (overlapping command) is indicated as an **Asynchronous command**.
- **Reset values (*RST)**
  Default parameter values that are used directly after resetting the instrument (`*RST` command) are indicated as ***RST** values, if available.
- **Default unit**
  This is the unit used for numeric values if no other unit is provided with the parameter.
- **Manual operation**
  If the result of a remote command can also be achieved in manual operation, a link to the description is inserted.

22.2 Programming Examples

The following section provide simple programming examples for the R&S Pulse Sequencer. The purpose of the examples is to present **all** commands for a given task. In real applications, one would rather reduce the examples to an appropriate subset of commands.

The programming examples have been tested with the built-in console which provides an environment for the development and execution of remote tests. To keep the examples as simple as possible, only the "clean" SCPI-like syntax elements are reported. Non-executable command lines (e.g. comments) start with two // characters.

22.3 Common Commands

Common commands are described in the IEEE 488.2 (IEC 625-2) standard. These commands have the same effect and are employed in the same way on different devices. The headers of these commands consist of "*" followed by three letters. Many common commands are related to the Status Reporting System.

Available common commands:

*IDN?....270
*OPT?....271
*RST....271

***IDN?**

Identification

Returns the instrument identification.

Return values:

<ID> "Rohde&Schwarz,<device type>,<serial number>,<firmware version>"

Example: `Rohde&Schwarz,K300,1407.6004k02/000000, 3.1.17.1-03.01.158`

Usage: Query only

***OPT?**

Option identification query

Queries the options included in the instrument. For a list of all available options and their description refer to the data sheet.

Return values:

<Options> The query returns a list of options. The options are returned at fixed positions in a comma-separated string. A zero is returned for options that are not installed.

Usage: Query only

***RST**

Reset

Sets the software to a defined default status. The default settings are indicated in the description of commands.

Unsaved setting are lost.

Usage: Setting only

22.4 Commands with Similar Syntax

This section describes command, that follow similar syntax and are common to the command groups listed in the following sections.

Commands for handling of repository elements

Repository elements are referenced by their *unique name*. The name may contain empty spaces. Almost In each of the command groups, there is a `...:SELect` command. Use this command to select one repository element to which the subsequent commands apply.

A typical remote control sequence would look like in the example "Handling repository elements" on page 271.

Example: Handling repository elements

The following is a simple example that explains the principle of repository elements handling by using the common commands.

```
// activate the remote control mode
SCPI
// create new empty repository
REPository:CREate "Repository for tests"
REPository:CATalog?
// "Repository for tests"
REPository:SAVE

// create two new scenarios
SCENario:CREate "New 1"
SCENario:CREate "New 2"
SCENario:SELect "New 1"
SCENario:COMMent "simple test scenario"
// rename the scenario
SCENario:NAME "New"
// remove it from the repository
SCENario:CATalog?
// "New", "New 2"
SCENario:REMove "New 2"
SCENario:CATalog?
// "New"

// create new pulse
// PULSe:CREate "New"
PULSe:CATalog?
// "New"
// although the pulse and the scenario use the same name,
// they are different and unique elements for the repository
```

Commands for handling of list and table items

Several repository elements are described in table form or as lists, like for example the sequence description table, the FM and AM Step modulations, the data source, IPM profiles in list form, lists of multiple emitters, etc.

Lists and tables are composed of items, where items are referenced by their *number*. Use the `...:SELect` command to select one item to which the subsequent commands apply.

A typical remote control sequence would look like in the example "Handling items" on page 272.

Example: Handling items

The following is a simple example that explains the principle of items handling by using the common commands.

```
// activate the remote control mode
SCPI
// creates a pulse-based sequence with 2 items (i.e. segments)
SEQuence:CREate "Seq1"
SEQuence:SELect "Seq1"
```

```
SEQuence:TYPE PULSe
SEQuence:ITEM:ADD
SEQuence:ITEM:COUNt?
// 1
SEQuence:ITEM:SELect 1; SEQuence:ITEM:TYPE PULSe
SEQuence:ITEM:PULSe "P1"
SEQuence:ITEM:ADD
SEQuence:ITEM:COUNt?
// 2
SEQuence:ITEM:DELete 2
SEQuence:ITEM:COUNt?
// 1
SEQuence:ITEM:CLEar
SEQuence:ITEM:COUNt?
// 0
```

ANTenna:CATalog?.....276
DSRC:CATalog?.....276
EMITter:CATalog?.....276
GENerator:CATalog?.....276
IPM:CATalog?.....276
PLUGin:CATalog?.....276
PULSe:CATalog?.....276
SCAN:CATalog?.....276
SCENario:CATalog?.....276
SEQuence:CATalog?.....276
WAVeform:CATalog?.....276
REPository:CATalog?.....276
ANTenna:CREate.....276
DSRC:CREate.....276
EMITter:CREate.....276
GENerator:CREate.....277
IPM:CREate.....277
PLUGin:CREate.....277
PULSe:CREate.....277
SCAN:CREate.....277
SCENario:CREate.....277
SEQuence:CREate.....277
WAVeform:CREate.....277
REPository:CREate.....277
ANTenna:SELect.....277
DSRC:SELect.....277
EMITter:SELect.....277
GENerator:SELect.....277
IPM:SELect.....277
PLUGin:SELect.....277
PULSe:SELect.....277
SCAN:SELect.....277
SCENario:SELect.....277
SEQuence:SELect.....277

WAVeform:SELect 277
REPository:SELect 277
ANTenna:NAME 277
DSRC:NAME 277
EMITter:MODE:NAME 277
EMITter:NAME 277
GENerator:NAME 277
IPM:NAME 278
IPM:PLUGin:NAME 278
PLUGin:MODule:NAME? 278
PLUGin:NAME 278
PULSe:MOP:PLUGin:NAME 278
PULSe:NAME 278
SCAN:NAME 278
SCENario:NAME 278
SEQuence:NAME 278
WAVeform:NAME 278
ANTenna:COMMent 278
DSRC:COMMent 278
EMITter:COMMent 278
GENerator:COMMent 278
IPM:COMMent 278
PLUGin:COMMent 278
PULSe:COMMent 278
PULSe:MOP:COMMent 278
SCAN:COMMent 278
SCENario:COMMent 278
SEQuence:COMMent 278
WAVeform:COMMent 278
REPository:COMMent 278
ANTenna:REMove 278
DSRC:REMove 278
EMITter:REMove 278
GENerator:REMove 278
IPM:REMove 278
PLUGin:REMove 278
PULSe:REMove 278
SCAN:REMove 279
SCENario:REMove 279
SEQuence:REMove 279
WAVeform:REMove 279
REPository:REMove 279
ASSignment:GENerator:PATH:EMITter:ADD 279
EMITter:MODE:ADD 279
EMITter:MODE:BEAM:ADD 279
PULSe:MOP:AMSTep:ADD 279
PULSe:MOP:FMSTep:ADD 279
PULSe:MOP:PCHirp:ADD 279
PULSe:MOP:PLISt:ADD 279
SCENario:LOCalized:ADD 279

SCENario:CSEQuence:ADD 279
SCENario:CEMit:ADD 279
DSRC:ITEM:ADD 279
IPM:LIST:ITEM:ADD 279
PULSe:ENVelope:DATA:ITEM:ADD 279
SEQuence:ITEM:IPM:ADD 279
SEQuence:ITEM:ADD 279
INSTrument:COUNt? 280
EMITter:MODE:BEAM:COUNt? 280
EMITter:MODE:COUNt? 280
PULSe:ENVelope:DATA:ITEM:COUNt? 280
PULSe:MOP:AMSTep:COUNt? 280
PULSe:MOP:FMSTep:COUNt? 280
PULSe:MOP:PCHirp:COUNt? 280
PULSe:MOP:PLISt:COUNt? 280
IPM:LIST:ITEM:COUNt? 280
SEQuence:ITEM:IPM:COUNt? 280
SEQuence:ITEM:COUNt? 280
INSTrument:SELect 280
ASSignment:EMITters:SELect 280
ASSignment:GENerator:PATH:EMITter:SELect 280
ASSignment:GENerator:PATH:SELect 280
ASSignment:GENerator:SELect 280
EMITter:MODE:BEAM:SELect 280
EMITter:MODE:SELect 280
PULSe:ENVelope:DATA:ITEM:SELect 280
PULSe:MOP:AMSTep:SELect 280
PULSe:MOP:FMSTep:SELect 280
PULSe:MOP:PCHirp:SELect 280
PULSe:MOP:PLISt:SELect 280
SCENario:LOCalized:SELect 280
SCENario:CSEQuence:SELect 280
SCENario:CEMit:SELect 280
DSRC:ITEM:SELect 280
IPM:LIST:ITEM:SELect 280
SEQuence:ITEM:IPM:SELect 280
SEQuence:ITEM:SELect 280
PULSe:MOP:AMSTep:INSert 281
PULSe:MOP:FMSTep:INSert 281
PULSe:MOP:PCHirp:INSert 281
PULSe:MOP:PLISt:INSert 281
INSTrument:DELete 281
ASSignment:GENerator:PATH:EMITter:DELete 281
DSRC:ITEM:DELete 281
EMITter:MODE:BEAM:DELete 281
EMITter:MODE:DELete 281
IPM:LIST:ITEM:DELete 281
PULSe:ENVelope:DATA:ITEM:DELete 281
PULSe:MOP:AMSTep:DELete 281
PULSe:MOP:FMSTep:DELete 281

PULSe:MOP:PCHirp:DELete....281
PULSe:MOP:PLISt:DELete....281
SCENario:LOCalized:DELete....281
SCENario:CSEQuence:DELete....281
SCENario:CEMit:DELete....281
SEQuence:ITEM:IPM:DELete....281
SEQuence:ITEM:DELete....281
INSTrument:CLEar....281
ANTenna:MODel:USER:CLEar....281
DSRC:CLEar....281
EMITter:MODE:ANTenna:CLEar....281
EMITter:MODE:BEAM:CLEar....281
EMITter:MODE:CLEar....282
EMITter:MODE:SCAN:CLEar....282
IPM:LIST:CLEar....282
PULSe:ENVelope:DATA:CLEar....282
PULSe:MOP:AMSTep:CLEar....282
PULSe:MOP:FMSTep:CLEar....282
PULSe:MOP:PCHirp:CLEar....282
PULSe:MOP:PLISt:CLEar....282
SCENario:EMITter:CLEar....282
SCENario:CSEQuence:CLEar....282
SEQuence:ITEM:CLEar....282

ANTenna:CATalog?
DSRC:CATalog?
EMITter:CATalog?
GENerator:CATalog?
IPM:CATalog?
PLUGin:CATalog?
PULSe:CATalog?
SCAN:CATalog?
SCENario:CATalog?
SEQuence:CATalog?
WAVeform:CATalog?
REPository:CATalog?

Queries the available repository elements in the database.

Return values:

<Catalog> string

Example: see example "Working with repositories" on page 350

Usage: Query only

Manual operation: See "General Repository Settings" on page 42

ANTenna:CREate <Create>
DSRC:CREate <Create>
EMITter:CREate <Create>

GENerator:CREate <Create>
IPM:CREate <Create>
PLUGin:CREate <Create>
PULSe:CREate <Create>
SCAN:CREate <Create>
SCENario:CREate <Create>
SEQuence:CREate <Create>
WAVeform:CREate <Create>
REPository:CREate <Create>

Creates a repository element with the selected name.

Setting parameters:

<Create> string
Must be unique for the particular type of repository elements. May contain empty spaces.

Example: see example "Working with repositories" on page 350

Usage: Setting only

Manual operation: See "General Repository Settings" on page 42

ANTenna:SELect <Select>
DSRC:SELect <Select>
EMITter:SELect <Select>
GENerator:SELect <Select>
IPM:SELect <Select>
PLUGin:SELect <Select>
PULSe:SELect <Select>
SCAN:SELect <Select>
SCENario:SELect <Select>
SEQuence:SELect <Select>
WAVeform:SELect <Select>
REPository:SELect <Select>

Selects the repository element to which the subsequent commands apply.

Parameters:

<Select> string
Element name, as defined with the `...:CREate` or `...:NAME` command.

Example: see example "Working with repositories" on page 350

Manual operation: See "General Repository Settings" on page 42

ANTenna:NAME <Name>
DSRC:NAME <Name>
EMITter:MODE:NAME <Name>
EMITter:NAME <Name>
GENerator:NAME <Name>

IPM:NAME <Name>
IPM:PLUGin:NAME <Name>
PLUGin:MODule:NAME?
PLUGin:NAME <Name>
PULSe:MOP:PLUGin:NAME <Name>
PULSe:NAME <Name>
SCAN:NAME <Name>
SCENario:NAME <Name>
SEQuence:NAME <Name>
WAVeform:NAME <Name>

Renames the selected repository element.

Parameters:

<Name> string

Must be unique for the particular type of repository elements. May contain empty spaces.

Example: see example "Handling repository elements" on page 271

Manual operation: See "Waveform Name" on page 193

ANTenna:COMMent <Comment>
DSRC:COMMent <Comment>
EMITter:COMMent <Comment>
GENerator:COMMent <Comment>
IPM:COMMent <Comment>
PLUGin:COMMent <Comment>
PULSe:COMMent <Comment>
PULSe:MOP:COMMent <Comment>
SCAN:COMMent <Comment>
SCENario:COMMent <Comment>
SEQuence:COMMent <Comment>
WAVeform:COMMent <Comment>
REPository:COMMent <Comment>

Adds a description to the selected repository element.

Parameters:

<Comment> string

Example: see example "Working with repositories" on page 350

Manual operation: See "Comment" on page 43

ANTenna:REMove <Remove>
DSRC:REMove <Remove>
EMITter:REMove <Remove>
GENerator:REMove <Remove>
IPM:REMove <Remove>
PLUGin:REMove <Remove>
PULSe:REMove <Remove>

SCAN:REMove <Remove>
SCENario:REMove <Remove>
SEQuence:REMove <Remove>
WAVeform:REMove <Remove>
REPository:REMove <File>[,<Username>,<Passwd>]

Removes the selected element from the workspace.

The element must not reference any child elements. Remove the referenced elements first.

Setting parameters:

<File> string
Element name, as defined with the `...:CREate` or `...:NAME` command.

<Username> string
required if the repository is password protected

<Passwd> string
required if the repository is password protected

Example: see example "Working with repositories" on page 350

Usage: Setting only

Manual operation: See "General Repository Settings" on page 42

ASSignment:GENerator:PATH:EMITter:ADD
EMITter:MODE:ADD
EMITter:MODE:BEAM:ADD
PULSe:MOP:AMSTep:ADD
PULSe:MOP:FMSTep:ADD
PULSe:MOP:PCHirp:ADD
PULSe:MOP:PLISt:ADD
SCENario:LOCalized:ADD
SCENario:CSEQuence:ADD
SCENario:CEMit:ADD
DSRC:ITEM:ADD
IPM:LIST:ITEM:ADD
PULSe:ENVelope:DATA:ITEM:ADD
SEQuence:ITEM:IPM:ADD
SEQuence:ITEM:ADD

Appends new item.

Example: see example "Handling items" on page 272

Usage: Event

Manual operation: See "New, Insert, Append, Remove, Clear Items" on page 109

INSTrument:COUNt?
EMITter:MODE:BEAM:COUNt?
EMITter:MODE:COUNt?
PULSe:ENVelope:DATA:ITEM:COUNt?
PULSe:MOP:AMSTep:COUNt?
PULSe:MOP:FMSTep:COUNt?
PULSe:MOP:PCHirp:COUNt?
PULSe:MOP:PLISt:COUNt?
IPM:LIST:ITEM:COUNt?
SEQuence:ITEM:IPM:COUNt?
SEQuence:ITEM:COUNt?

Querries the number of existing items.

Return values:
<Count> integer
*RST: 0

Example: see example "Handling items" on page 272

Usage: Query only

Manual operation: See "Select, No." on page 110

INSTrument:SELect <Select>
ASSignment:EMITters:SELect <Select>
ASSignment:GENerator:PATH:EMITter:SELect <Select>
ASSignment:GENerator:PATH:SELect <Select>
ASSignment:GENerator:SELect <Select>
EMITter:MODE:BEAM:SELect <Select>
EMITter:MODE:SELect <Select>
PULSe:ENVelope:DATA:ITEM:SELect <Select>
PULSe:MOP:AMSTep:SELect <Select>
PULSe:MOP:FMSTep:SELect <Select>
PULSe:MOP:PCHirp:SELect <Select>
PULSe:MOP:PLISt:SELect <Select>
SCENario:LOCalized:SELect <Select>
SCENario:CSEQuence:SELect <Select>
SCENario:CEMit:SELect <Select>
DSRC:ITEM:SELect <Select>
IPM:LIST:ITEM:SELect <Select>
SEQuence:ITEM:IPM:SELect <Select>
SEQuence:ITEM:SELect <Select>

Selects the item to which the subsequent commands apply.

Parameters:
<Select> float
Item number within the range 1 .. `....:COUNt`
Range: 1 to 4096

Example: see example "Handling items" on page 272

Manual operation: See "Select, No." on page 110

PULSe:MOP:AMSTep:INSert <Insert>
PULSe:MOP:FMSTep:INSert <Insert>
PULSe:MOP:PCHirp:INSert <Insert>
PULSe:MOP:PLISt:INSert <Insert>

Inserts a new item before the selected one.

Setting parameters:
<Insert> float
*RST: 0

Usage: Setting only

Manual operation: See "Custom Phase" on page 90

INSTrument:DELete <Delete>
ASSignment:GENerator:PATH:EMITter:DELete
DSRC:ITEM:DELete <Delete>
EMITter:MODE:BEAM:DELete <Delete>
EMITter:MODE:DELete <Delete>
IPM:LIST:ITEM:DELete <Delete>
PULSe:ENVelope:DATA:ITEM:DELete <Delete>
PULSe:MOP:AMSTep:DELete <Delete>
PULSe:MOP:FMSTep:DELete <Delete>
PULSe:MOP:PCHirp:DELete <Delete>
PULSe:MOP:PLISt:DELete <Delete>
SCENario:LOCalized:DELete <Delete>
SCENario:CSEQuence:DELete <Delete>
SCENario:CEMit:DELete <Delete>
SEQuence:ITEM:IPM:DELete <Delete>
SEQuence:ITEM:DELete <Delete>

Deletes the particular item.

Setting parameters:
<Delete> float
*RST: 0

Example: see example "Handling items" on page 272

Usage: Setting only

Manual operation: See "New, Insert, Append, Remove, Clear Items" on page 109

INSTrument:CLEar
ANTenna:MODel:USER:CLEar
DSRC:CLEar
EMITter:MODE:ANTenna:CLEar
EMITter:MODE:BEAM:CLEar

EMITter:MODE:CLEar
EMITter:MODE:SCAN:CLEar
IPM:LIST:CLEar
PULSe:ENVelope:DATA:CLEar
PULSe:MOP:AMSTep:CLEar
PULSe:MOP:FMSTep:CLEar
PULSe:MOP:PCHirp:CLEar
PULSe:MOP:PLISt:CLEar
SCENario:EMITter:CLEar
SCENario:CSEQuence:CLEar
SEQuence:ITEM:CLEar

Deletes all items from the list or the table.

Example: see example "Handling items" on page 272

Usage: Event

Manual operation: See "New, Insert, Append, Remove, Clear Items" on page 109

22.5 Antenna Pattern Commands

Example: Configuring antenna patterns

```
SCPI
ANTenna:CREate "My_PlanarAntenna"
ANTenna:COMMent "planar phased antenna with parabolic aperture distribution"
ANTenna:MODel:TYPE ARRay
ANTenna:MODel:FREQuency 1e+10
ANTenna:MODel:BANDwidth 1e+9
ANTenna:MODel:ARRay:DISTribution PARabolic
ANTenna:MODel:ARRay:NX 12
ANTenna:MODel:ARRay:NZ 12
ANTenna:MODel:ARRay:XDIStance 0.6
ANTenna:MODel:ARRay:ZDIStance 0.3
ANTenna:MODel:ARRay:PEDestal 0.1

ANTenna:CREate "Test"
ANTenna:MODel:TYPE HORN
ANTenna:MODel:ROTation:Z 0
ANTenna:MODel:ROTation:X 0
ANTenna:MODel:FREQuency 1e+10
ANTenna:MODel:HORN:LX 0.076
ANTenna:MODel:HORN:LZ 0.05

ANTenna:CREate "My_PencilBeam"
ANTenna:MODel:TYPE SINC
ANTenna:MODel:ROTation:Z 0
ANTenna:MODel:ROTation:X 0
ANTenna:MODel:SINC:HPBW 3

ANTenna:CREate "Test CSC"
ANTenna:MODel:TYPE COSecant
ANTenna:MODel:COSecant:HPBW 2
ANTenna:MODel:COSecant:T1 5
ANTenna:MODel:COSecant:T2 70

ANTenna:CREate "User"
ANTenna:MODel:TYPE USER
ANTenna:MODel:USER:LOAD "C:/_PS_files/antenna.ant_pat"
ANTenna:MODel:ROTation:Z -90
ANTenna:MODel:ROTation:X 0
ANTenna:MODel:USER:CLEAr

ANTenna:CREate "Custom"
ANTenna:MODel:TYPE CUSTom
ANTenna:MODel:CUSTom:HPBW:XY 10
ANTenna:MODel:CUSTom:HPBW:YZ 3
ANTenna:MODel:CUSTom:SLSTart 30
```

```
ANTenna:MODel:CUSTom:SLRolloff 10
ANTenna:MODel:CUSTom:SLSCale 0.5
ANTenna:MODel:BACKlobe:ENABle 1
ANTenna:MODel:BACKlobe:TYPE MIRRor
ANTenna:MODel:BACKlobe:ATTenuation 40
```

ANTenna:MODel:ARRay:COSN....284
ANTenna:MODel:ARRay:DISTribution....285
ANTenna:MODel:ARRay:NX....285
ANTenna:MODel:ARRay:NZ....285
ANTenna:MODel:ARRay:PEDestal....285
ANTenna:MODel:ARRay:XDIStance....285
ANTenna:MODel:ARRay:ZDIStance....285
ANTenna:MODel:BACKlobe:ENABle....285
ANTenna:MODel:BACKlobe:TYPE....286
ANTenna:MODel:BACKlobe:ATTenuation....286
ANTenna:MODel:COSecant:T1....286
ANTenna:MODel:COSecant:T2....286
ANTenna:MODel:CUSTom:HPBW:XY....286
ANTenna:MODel:CUSTom:HPBW:YZ....286
ANTenna:MODel:CUSTom:SLSTart....287
ANTenna:MODel:CUSTom:SLRolloff....287
ANTenna:MODel:CUSTom:SLSCale....287
ANTenna:MODel:BANDwidth....287
ANTenna:MODel:FREQuency....287
ANTenna:MODel:GAUSsian:HPBW....288
ANTenna:MODel:COSecant:HPBW....288
ANTenna:MODel:SINC:HPBW....288
ANTenna:MODel:HORN:LX....288
ANTenna:MODel:HORN:LZ....288
ANTenna:MODel:PARabolic:DIAMeter....288
ANTenna:MODel:ROTation:X....289
ANTenna:MODel:ROTation:Z....289
ANTenna:MODel:TYPE....289
ANTenna:MODel:USER:LOAD....289
ANTenna:MODel:USER:CSV:FORMat....289
PLOT:POLar:TYPE....290
PLOT:POLar:CUT....290
PLOT:POLar:LOG:MIN....290

ANTenna:MODel:ARRay:COSN <Cosn>

Sets Cos^N of the Planar Phased Array antenna.

Parameters:

<Cosn> float
Range: 2 to 10

Example: example "Configuring antenna patterns" on page 283

Manual operation: See "Planar Phased Array Antenna Settings" on page 160

ANTenna:MODel:ARRay:DISTribution <Distribution>

Sets the aperture distribution of the Planar Phased Array antenna.

Parameters:

<Distribution> UNIForm | PARabolic | COSine | CSQuared | COSN | TRIangular

Example: example "Configuring antenna patterns" on page 283

Manual operation: See "Planar Phased Array Antenna Settings" on page 160

ANTenna:MODel:ARRay:NX <Nx>
ANTenna:MODel:ARRay:NZ <Nz>

Sets the number of elements of the Planar Phased Array antenna.

Parameters:

<Nz> float

Range: 1 to 1000

Example: example "Configuring antenna patterns" on page 283

Manual operation: See "Planar Phased Array Antenna Settings" on page 160

ANTenna:MODel:ARRay:PEDestal <Pedestal>

Sets the Pedestal of the Planar Phased Array antenna.

Parameters:

<Pedestal> float

Range: 0 to 1

Example: example "Configuring antenna patterns" on page 283

Manual operation: See "Planar Phased Array Antenna Settings" on page 160

ANTenna:MODel:ARRay:XDIStance <Xdistance>
ANTenna:MODel:ARRay:ZDIStance <Zdistance>

Sets the X, Z elements of the Planar Phased Array antenna.

Parameters:

<Zdistance> float

Range: 0.0001 to 1

Example: example "Configuring antenna patterns" on page 283

Manual operation: See "Planar Phased Array Antenna Settings" on page 160

ANTenna:MODel:BACKlobe:ENABle <Enable>

Enables the simulation of a back lobe.

Parameters:
<Enable> ON | OFF | 1 | 0

Example: example "Configuring antenna patterns" on page 283

Manual operation: See "Simulate Back Lobe, Attenuation, Type " on page 157

ANTenna:MODel:BACKlobe:TYPE <Type>

Sets the shape of the back lobe pattern.

Parameters:
<Type> MIRRor | OMNidirect

Example: example "Configuring antenna patterns" on page 283

Manual operation: See "Simulate Back Lobe, Attenuation, Type " on page 157

ANTenna:MODel:BACKlobe:ATTenuation <Attenuation>

Sets the attenuation of the back lobe.

Parameters:
<Attenuation> float
Range: 0 to 100

Example: example "Configuring antenna patterns" on page 283

Manual operation: See "Simulate Back Lobe, Attenuation, Type " on page 157

ANTenna:MODel:COSecant:T1 <T1>
ANTenna:MODel:COSecant:T2 <T2>

Sets the Theta parameters.

Parameters:
<T2> float
Range: 1 to 90

Example: example "Configuring antenna patterns" on page 283

Manual operation: See "Cosecant Squared Antenna Settings" on page 160

ANTenna:MODel:CUSTom:HPBW:XY <Xy>
ANTenna:MODel:CUSTom:HPBW:YZ <Yz>

Sets the required HPBW of the custom antenna.

Parameters:
<Yz> float
Range: 1 to 90

Example: example "Configuring antenna patterns" on page 283

Manual operation: See "Custom Antenna Settings" on page 161

ANTenna:MODel:CUSTom:SLSTart <Slstart>

Sets the power level of the first pairs of side lobes.

Parameters:
<Slstart> float
Range: 1 to 90

Example: example "Configuring antenna patterns" on page 283

Manual operation: See "Custom Antenna Settings" on page 161

ANTenna:MODel:CUSTom:SLRolloff <Slrolloff>

Sets the factor used to calculate the HPBW of the side lobes.

Parameters:
<Slrolloff> float
Range: 1 to 90

Example: example "Configuring antenna patterns" on page 283

Manual operation: See "Custom Antenna Settings" on page 161

ANTenna:MODel:CUSTom:SLSCale <Slscale>

Sets the step size to calculate the power level of the side lobes.

Parameters:
<Slscale> float
Range: 0.01 to 10

Example: example "Configuring antenna patterns" on page 283

Manual operation: See "Custom Antenna Settings" on page 161

ANTenna:MODel:BANDwidth <Bandwidth>

Sets the antenna bandwidth.

Parameters:
<Bandwidth> float
Range: 1e+06 to 1e+11

Example: example "Configuring antenna patterns" on page 283

Manual operation: See "Frequency, Bandwidth" on page 157

ANTenna:MODel:FREQuency <Frequency>

Sets the freqeuncy.

Parameters:

<Frequency> float

Range: 1e+06 to 1e+11
Default unit: Hz

Example: example "Configuring antenna patterns" on page 283

Manual operation: See "Frequency, Bandwidth" on page 157

ANTenna:MODel:GAUSsian:HPBW <Hpbw>
ANTenna:MODel:COSecant:HPBW <Hpbw>
ANTenna:MODel:SINC:HPBW <Hpbw>

Sets the Half-Power Beam Width for the Gaussian, Sin(x)/x and Cosecant Squared antennas.

Parameters:

<Hpbw> float

Range: 0.1 to 30
Default unit: degree

Example: example "Configuring antenna patterns" on page 283

Manual operation: See "Sin(x)/x Antenna Settings" on page 159

ANTenna:MODel:HORN:LX <Lx>
ANTenna:MODel:HORN:LZ <Lz>

Sets the length of the rectangular sides of the Pyramidal Horn antenna.

Parameters:

<Lz> float

Range: 0.01 to 100
Default unit: m

Example: example "Configuring antenna patterns" on page 283

Manual operation: See "Pyramidal Horn Antenna Settings" on page 160

ANTenna:MODel:PARabolic:DIAMeter <Diameter>

Sets the diameter of the parabolic dish antenna.

Parameters:

<Diameter> float

Range: 0.05 to 100
Default unit: m

Example: example "Configuring antenna patterns" on page 283

Manual operation: See "Parabolic Antenna Settings" on page 158

ANTenna:MODel:ROTation:X <X>
ANTenna:MODel:ROTation:Z <Z>

Sets the X and Z antenna rotation.

Parameters:
<Z> float
Range: -180 to 180
Default unit: degree

Example: example "Configuring antenna patterns" on page 283

Manual operation: See "Z-Rotation, X-Rotation" on page 157

ANTenna:MODel:TYPE <Type>

Sets the antenna pattern.

Parameters:
<Type> DIPole | PARabolic | GAUSsian | SINC | HORN | COSecant | ARRay | USER | CUSTom

Example: example "Configuring antenna patterns" on page 283

Manual operation: See "Antenna Model" on page 156

ANTenna:MODel:USER:LOAD <Load>

Loads a custom antenna pattern file.

Setting parameters:
<Load> string

Example: example "Configuring antenna patterns" on page 283

Usage: Setting only

Manual operation: See "Imported Antenna Settings" on page 161

ANTenna:MODel:USER:CSV:FORMat
<start_row>,<col_sep>,<decimal>,<col>{,<col>}

Defines how the data in the selected `*.csv` file should be interpreted.

The settings in this command are not permanent. The command affects only the currently selected antenna.

For description of the `*.csv` file format, see chapter A.1.2, "Antenna Pattern File Formats", on page 393.

Parameters:
<col_sep> COMMA | SEMICOLON | WHITESPACE
Column separator

<decimal> POINT | COMMA
Decimal delimiter

<col> THETA | PHI | GAIN | ETHRE | ETHIM | EPHRE | EPHIM
Data content of the column: Theta, Phi, Gain, Re(E_Theta), Im(E_Theta), Re(E_Phi), Im(E_Phi)
At least one out of maximum 6 columns should exist

Setting parameters:
<start_row> First row with data
Range: 1 to max

Example:
```
ANTenna:MODel:TYPE USER
ANTenna:MODel:USER:CSV:FORMat "2","COMMA","POINT","PHI","THETA","GAIN"
ANTenna:MODel:USER:LOAD "D:/PS/ansys.csv"
```

Usage: Setting only

Manual operation: See "Import Wizard" on page 162

PLOT:POLar:TYPE <Type>

Sets the coordinates of the 2D antenna pattern diagram.

Setting parameters:
<Type> POLar | CARTesian

Example:
```
PLOT:POLar:TYPE CARTesian
PLOT:POLar:CUT XY
PLOT:POLar:LOG:MIN -90
```

Usage: Setting only

Manual operation: See "2D and 3D diagrams" on page 158

PLOT:POLar:CUT <Cut>

Sets the diagram cut.

Setting parameters:
<Cut> XY | YZ

Example: see `PLOT:POLar:TYPE` on page 290

Usage: Setting only

Manual operation: See "2D and 3D diagrams" on page 158

PLOT:POLar:LOG:MIN <Min>

Sets the minimum value displayed on the y axis.

Setting parameters:
<Min> float

Example: see PLOT:POLar:TYPE on page 290

Usage: Setting only

Manual operation: See "2D and 3D diagrams" on page 158

22.6 Antenna Scan Commands

Example: Defining antenna scans

```
SCPI
SCAN:CREate "My_RasterScan"
SCAN:TYPE RASTer
SCAN:RASTer:WIDTh 70
SCAN:RASTer:BARWidth 3
SCAN:RASTer:RATE 15
SCAN:RASTer:BARS 10
SCAN:RASTer:DIRection HORizontal
SCAN:RASTer:RETRace 0.001
SCAN:RASTer:UNIDirection 1
SCAN:RASTer:FLYBack 0.001
SCAN:RASTer:PALMer 1
SCAN:RASTer:PRATe 5
SCAN:RASTer:PSQuint 1.5

SCAN:CREate "Test Antenna Scan"
SCAN:TYPE HELical
SCAN:HELical:RPM 45
SCAN:HELical:TURNs 8
SCAN:HELical:ELEVation:STEP 3
SCAN:HELical:RETRace 1
SCAN:HELical:ROTation CW
```

SCAN:CIRCular:ROTation 292
SCAN:CONical:ROTation 292
SCAN:HELical:ROTation 292
SCAN:LSW:ROTation 292
SCAN:SPIRal:ROTation 292
SCAN:CIRCular:RPM 293
SCAN:HELical:RPM 293
SCAN:CONical:RATE 293
SCAN:SECTor:RATE 293
SCAN:RASTer:RATE 293
SCAN:HELical:ELEVation:STEP 293
SCAN:HELical:TURNs 293
SCAN:LSW:DIRection 294
SCAN:LSW:DWELl 294
SCAN:LSW:LOBes 294
SCAN:CONical:SQUint 294

SCAN:LSW:SQUint....294
SCAN:RASTer:BARS....294
SCAN:RASTer:BARWidth....295
SCAN:RASTer:FLYBack....295
SCAN:SECTor:FLYBack....295
SCAN:HELical:RETRace....295
SCAN:RASTer:RETRace....295
SCAN:SPIRal:RETRace....295
SCAN:RASTer:UNIDirection....295
SCAN:SECTor:UNIDirection....295
SCAN:RASTer:WIDTh....296
SCAN:SECTor:WIDTh....296
SCAN:RASTer:DIRection....296
SCAN:SPIRal:ROUNds....296
SCAN:SPIRal:RTIMe....296
SCAN:SPIRal:STEP....297
SCAN:TYPE....297
SCAN:CIRCular:NODDing....297
SCAN:SECTor:NODDing....297
SCAN:CIRCular:NELevation....297
SCAN:SECTor:NELevation....297
SCAN:CIRCular:NRATe....298
SCAN:SECTor:NRATe....298
SCAN:CIRCular:PALMer....298
SCAN:RASTer:PALMer....298
SCAN:SECTor:PALMer....298
SCAN:CIRCular:PRATe....298
SCAN:RASTer:PRATe....298
SCAN:SECTor:PRATe....298
SCAN:CIRCular:PSQuint....298
SCAN:RASTer:PSQuint....298
SCAN:SECTor:PSQuint....298

SCAN:CIRCular:ROTation <Rotation>
SCAN:CONical:ROTation <Rotation>
SCAN:HELical:ROTation <Rotation>
SCAN:LSW:ROTation <Rotation>
SCAN:SPIRal:ROTation <Rotation>

Sets the rotation direction of the antenna.

Parameters:

<Rotation> CW | CCW

Example: see example "Defining antenna scans" on page 291

Manual operation: See "Spiral Scan Settings" on page 169

SCAN:CIRCular:RPM <Rpm>
SCAN:HELical:RPM <Rpm>

Sets the rotation speed of the antenna.

Parameters:
<Rpm> float
Range: 0.01 to 1000
Default unit: degree/s

Example: example "Defining antenna scans" on page 291

Manual operation: See "Helical Scan Settings" on page 169

SCAN:CONical:RATE <Rate>
SCAN:SECTor:RATE <Rate>
SCAN:RASTer:RATE <Rate>

Sets the turning speed.

Parameters:
<Rate> float
Range: 0.01 to 1000
Default unit: degree/s

Example: example "Defining antenna scans" on page 291

Manual operation: See "Raster Scan Settings" on page 167

SCAN:HELical:ELEVation:STEP <Step>

Sets the step width the antenna is changing.

Parameters:
<Step> float
Range: 0.01 to 11.25
Default unit: degree

Example: example "Defining antenna scans" on page 291

Manual operation: See "Helical Scan Settings" on page 169

SCAN:HELical:TURNs <Turns>

Sets the number of turns.

Parameters:
<Turns> float
Range: 1 to 30

Example: example "Defining antenna scans" on page 291

Manual operation: See "Helical Scan Settings" on page 169

SCAN:LSW:DIRection <Direction>

Sets the horizontal or vertical switching direction.

Parameters:
<Direction> H | V

Example: example "Defining antenna scans" on page 291

Manual operation: See "Lobe Switching Scan Settings" on page 170

SCAN:LSW:DWELl <Dwell>

Sets the speed the antenna switches between the lobes.

Parameters:
<Dwell> float
Range: 1e-06 to 1

Example: example "Defining antenna scans" on page 291

Manual operation: See "Lobe Switching Scan Settings" on page 170

SCAN:LSW:LOBes <Lobes>

Set the number of lobes.

Parameters:
<Lobes> 2 | 4

Example: example "Defining antenna scans" on page 291

Manual operation: See "Lobe Switching Scan Settings" on page 170

SCAN:CONical:SQUint <Squint>
SCAN:LSW:SQUint <Squint>

Sets the offset angle of the antenna beam, that means for the conical antenna the parameter sets the radius of the scanned circle.

Parameters:
<Squint> float
Range: 0.05 to 15
Default unit: degree

Example: example "Defining antenna scans" on page 291

Manual operation: See "Lobe Switching Scan Settings" on page 170

SCAN:RASTer:BARS <Bars>

Sets the number of scanned bars (sectors).

Parameters:
<Bars> float
Range: 1 to 30

Example: example "Defining antenna scans" on page 291

Manual operation: See "Raster Scan Settings" on page 167

SCAN:RASTer:BARWidth <Barwidth>

Sets the distance between two consecutive scanned bars (sectors).

Parameters:
<Barwidth> float
Range: 0.1 to 9
Default unit: m

Example: example "Defining antenna scans" on page 291

Manual operation: See "Raster Scan Settings" on page 167

SCAN:RASTer:FLYBack <Flyback>
SCAN:SECTor:FLYBack <Flyback>

Sets the Flyback time for the antenna working in unidirectional mode.

Parameters:
<Flyback> float
Range: 0 to 1
Default unit: s

Example: example "Defining antenna scans" on page 291

Manual operation: See "Sector Scan Settings" on page 166

SCAN:HELical:RETRace <Retrace>
SCAN:RASTer:RETRace <Retrace>
SCAN:SPIRal:RETRace <Retrace>

Sets the speed for the antenna to return to the initial orientation.

Parameters:
<Retrace> float
Range: 0 to 1

Example: example "Defining antenna scans" on page 291

Manual operation: See "Spiral Scan Settings" on page 169

SCAN:RASTer:UNIDirection <Unidirection>
SCAN:SECTor:UNIDirection <Unidirection>

Enables an unidirectional scan mode.

Parameters:
<Unidirection> ON | OFF | 1 | 0

Example: example "Defining antenna scans" on page 291

Manual operation: See "Sector Scan Settings" on page 166

SCAN:RASTer:WIDTh <Width>
SCAN:SECTor:WIDTh <Width>

Sets the width of the sector to be scanned.

Parameters:
<Width> float
Range: 0.1 to 180
Default unit: degree

Example: example "Defining antenna scans" on page 291

Manual operation: See "Sector Scan Settings" on page 166

SCAN:RASTer:DIRection <Direction>

Sets the scanning direction.

Parameters:
<Direction> HORizontal | VERTical

Manual operation: See "Raster Scan Settings" on page 167

see example "Defining antenna scans" on page 291

SCAN:SPIRal:ROUNds <Rounds>

Sets the number of rounds the antenna performs.

Parameters:
<Rounds> float
Range: 0.1 to 15

Example: example "Defining antenna scans" on page 291

Manual operation: See "Spiral Scan Settings" on page 169

SCAN:SPIRal:RTIMe <Rtime>

Sets the turning speed of the antenna.

Parameters:
<Rtime> float
Range: 0.01 to 10
Default unit: degree/s

Example: example "Defining antenna scans" on page 291

Manual operation: See "Spiral Scan Settings" on page 169

SCAN:SPIRal:STEP <Step>

Determines the step size to increase the scan radius.

Parameters:
<Step> float
Range: 1 to 11.25
Default unit: degree

Example: example "Defining antenna scans" on page 291

Manual operation: See "Spiral Scan Settings" on page 169

SCAN:TYPE <Type>

Sets the scan type.

Parameters:
<Type> CIRCular | SECTor | RASTer | CONical | HELical | SPIRal | LSW

Example: example "Defining antenna scans" on page 291

Manual operation: See "Scan Type" on page 165

SCAN:CIRCular:NODDing <Nodding>
SCAN:SECTor:NODDing <Nodding>

Enables superimposing a horizontal nodding on the scan.

Parameters:
<Nodding> ON | OFF | 1 | 0

Example:
```
SCAN:SECTor:NODDing 1
SCAN:SECTor:NRATe 500
SCAN:SECTor:NELevation 15
```

Manual operation: See "Sector Scan Settings" on page 166

SCAN:CIRCular:NELevation <Nelevation>
SCAN:SECTor:NELevation <Nelevation>

Sets the elevation angle.

Parameters:
<Nelevation> float
Range: 0.01 to 90

Example: see `SCAN:SECTor:NODDing` on page 297

Manual operation: See "Sector Scan Settings" on page 166

SCAN:CIRCular:NRATe <Nrate>
SCAN:SECTor:NRATe <Nrate>

Sets the elevation rate.

Parameters:
<Nrate> float
Range: 0.01 to 2000

Example: see `SCAN:SECTor:NODDing` on page 297

Manual operation: See "Sector Scan Settings" on page 166

SCAN:CIRCular:PALMer <Palmer>
SCAN:RASTer:PALMer <Palmer>
SCAN:SECTor:PALMer <Palmer>

Enables superimposing a conical scan on the current scan.

Parameters:
<Palmer> ON | OFF | 1 | 0

Manual operation: See "Sector Scan Settings" on page 166

see example "Defining antenna scans" on page 291

SCAN:CIRCular:PRATe <Prate>
SCAN:RASTer:PRATe <Prate>
SCAN:SECTor:PRATe <Prate>

Sets the scan rate.

Parameters:
<Prate> float
Range: 0.1 to 1000

Manual operation: See "Sector Scan Settings" on page 166

see example "Defining antenna scans" on page 291

SCAN:CIRCular:PSQuint <Psquint>
SCAN:RASTer:PSQuint <Psquint>
SCAN:SECTor:PSQuint <Psquint>

Sets the squint angle.

Parameters:
<Psquint> float
Range: 0.05 to 15

Manual operation: See "Sector Scan Settings" on page 166

see example "Defining antenna scans" on page 291

22.7 Data Source Commands

Example: Creating data source

```
SCPI
DSRC:CREate "DataSource"
DSRC:SELect "DataSource"
DSRC:ITEM:ADD
DSRC:ITEM:SELect 1
DSRC:ITEM:TYPE PATTern
DSRC:ITEM:PATTern ONE
DSRC:ITEM:BITS 2

DSRC:ITEM:ADD
DSRC:ITEM:SELect 2
DSRC:ITEM:TYPE PRBS
DSRC:ITEM:PRBS:TYPE P15
DSRC:ITEM:PRBS:INIT 1
DSRC:ITEM:PRBS:INIT:VALue 10
DSRC:ITEM:BITS 12
```

DSRC:ITEM:BITS 299
DSRC:ITEM:DATA 299
DSRC:ITEM:PATTern 300
DSRC:ITEM:PRBS:INIT 300
DSRC:ITEM:PRBS:INIT:VALue 300
DSRC:ITEM:PRBS:TYPE 300
DSRC:ITEM:TYPE 301

DSRC:ITEM:BITS <Bits>

Sets the length of the selected item in bits.

Parameters:

<Bits> float
Range: 0 to 4096

Example: see example "Creating data source" on page 299

Manual operation: See "Data Source Table" on page 229

DSRC:ITEM:DATA <Data>

Parameters:

<Data> string

Manual operation: See "Data Source Table" on page 229

DSRC:ITEM:PATTern <Pattern>

Sets the data pattern of the selected item.

Parameters:

<Pattern> ZERO | ONE | ALT | R3 | R4A | R4B | R5 | R7 | R11 | R13

ZERO|ONE
Binary 0 and 1

ALT
Variable bit strings ("1010") with alternating 0 and 1 and a maximum length of 999 bits

R3|R4A|R4B|R5|R7|R11|R13
Barker codes

Example: see example "Creating data source" on page 299

Manual operation: See "Data Source Table" on page 229

DSRC:ITEM:PRBS:INIT <Init>

Enables/disables initialization of the shift register with a user-defined value.

Parameters:

<Init> ON | OFF

Example: see example "Creating data source" on page 299

Manual operation: See "Shift register" on page 229

DSRC:ITEM:PRBS:INIT:VALue <Value>

Set a new initialization value.

Parameters:

<Value> float

Range: 1 to 511

Example: see example "Creating data source" on page 299

Manual operation: See "Shift register" on page 229

DSRC:ITEM:PRBS:TYPE <Type>

Sets the PRBS type for the selected item.

Parameters:

<Type> P9 | P11 | P15 | P16 | P20 | P21 | P23 | P7

Example: see example "Creating data source" on page 299

Manual operation: See "Data Source Table" on page 229

DSRC:ITEM:TYPE <Type>

Sets the data type of selected item.

Parameters:

<Type> PATTern | PRBS | USER

Example: see example "Creating data source" on page 299

Manual operation: See "Data Source Table" on page 229

22.8 Emitter to Generator Mapping Commands

Example: Performing signal to generator mapping

```
SCPI
ASSignment:GENerator:LIST?
// My_RS_SMBV
// RS_SMW (Connected)
// My_RS_SMW
ASSignment:GENerator:SELect "My_RS_SMW"
ASSignment:GENerator:CAPabilities?
// RFA=2e+10, BWA=1.6e+08, MEMA=1e+09, SWA=K300+K301,
// RFB=6e+09, BWB=1.2e+08, MEMB=6.4e+07, SWB=K300
ASSignment:GENerator:PATH:LIST?
ASSignment:GENerator:PATH:SELect "Path A"
ASSignment:GENerator:PATH:EMITter:LIST?
// I1
// E3
ASSignment:EMITters:LIST?
// E1
// E2
ASSignment:EMITters:SELect "E2"
ASSignment:GENerator:PATH:SELect "Path A"
ASSignment:GENerator:PATH:EMITter:ADD
ASSignment:GENerator:PATH:EMITter:LIST?
// I1
// E2
// E3
ASSignment:GENerator:PATH:EMITter:SEL "I1"
ASSignment:GENerator:PATH:EMITter:DELete
ASSignment:GENerator:PATH:EMITter:LIST?
// E2
// E3
ASSignment:EMITters:LIST?
// I1
// E1
```

ASSignment:EMITters:LIST?....302
ASSignment:GENerator:LIST?....302
ASSignment:GENerator:CAPabilities?....302
ASSignment:GENerator:PATH:LIST?....302
ASSignment:GENerator:PATH:EMITter:LIST?....303

ASSignment:EMITters:LIST?

Queries the alias names of the available emitters/interfereres.

Return values:
<List> "<Emitter/Inter#1>","<Emitter/Inter#1#2>",...

Example: see example "Performing signal to generator mapping" on page 301

Usage: Query only

Manual operation: See "Emitters/Interferers" on page 251

ASSignment:GENerator:LIST?

Queries a list of the available generator profiles.

Return values:
<List> "<GenProfile#1>","<GenProfile#2>",...

Example: see example "Performing signal to generator mapping" on page 301

Usage: Query only

Manual operation: See "Generator Profiles" on page 251

ASSignment:GENerator:CAPabilities?

Queries the capabilites of the selected generator profile.

Return values:
<Capabilities> <RFA>, <BWA>, <MEMA>, <SWA>, [<RFB>, <BWB>, <MEMB>, <SWB>]

String that lists the maximum RF and BW, available memory size and installed options per path.

Example: see example "Performing signal to generator mapping" on page 301

Usage: Query only

Manual operation: See "Generator Profiles" on page 251

ASSignment:GENerator:PATH:LIST?

Queries the available paths.

Return values:

<List> "<Path#1>","<Path#2>",...
list of available paths

Example: see example "Performing signal to generator mapping" on page 301

Usage: Query only

Manual operation: See "Generator Profiles" on page 251

ASSignment:GENerator:PATH:EMITter:LIST?

Queries the list of currently assigned emitters/interferes to the selected path.

Return values:

<List> string

Example: see example "Performing signal to generator mapping" on page 301

Usage: Query only

Manual operation: See "Generator Profiles" on page 251

22.9 Emitter Commands

Example: Creating emitters

```
EMITter:CREate "My_EmitterGuidance"
EMITter:COMMent "guidance&survaillance"
EMITter:EIRP 120
EMITter:FREQuency 3e+09
EMITter:MODE:ADD
EMITter:MODE:SELect 1
EMITter:MODE:NAME "Guidance"
ANTenna:CATalog?
// "Isotropic","My_PencilBeam","Test CSC","My_PlanarAntenna","Test","Custom"
EMITter:MODE:ANTenna "My_PencilBeam"
SCAN:CATalog?
// "Test Antenna Scan","My_RasterScan"
EMITter:MODE:SCAN "My_RasterScan"
EMITter:MODE:BEAM:ADD
EMITter:MODE:BEAM:SELect 1
EMITter:MODE:BEAM:STATe 1
SEQuence:CATalog?
// "Test Sequence","My_S2_S1-F1_S1-F2","PT"
EMITter:MODE:BEAM:SEQuence "My_S2_S1-F1_S1-F2"
EMITter:MODE:BEAM:OFFSet:FREQuency 0
EMITter:MODE:BEAM:OFFSet:ELEVation 0
EMITter:MODE:BEAM:OFFSet:AZIMuth 0
```

EMITter:EIRP 304
EMITter:FREQuency 305
EMITter:MODE:ANTenna 305
EMITter:MODE:BEAM:OFFSet:AZIMuth 305
EMITter:MODE:BEAM:OFFSet:ELEVation 305
EMITter:MODE:BEAM:OFFSet:FREQuency 305
EMITter:MODE:BEAM:SEQuence 305
EMITter:MODE:BEAM:STATe 306
EMITter:MODE:SCAN 306

EMITter:EIRP <Eirp>

Sets the EIRP of the emitter.

Parameters:

<Eirp> float

Range: -100 to 200
Default unit: dBW

Example: see example "Creating emitters" on page 304

Manual operation: See "EIRP" on page 181

EMITter:FREQuency <Frequency>

Sets the operating frequency.

Parameters:
<Frequency> float
Range: 1000 to 4.4e+10
Default unit: Hz

Example: example "Creating emitters" on page 304

Manual operation: See "Frequency" on page 181

EMITter:MODE:ANTenna <Antenna>

Assigns an existing antenna pattern, see `ANTenna:CREate`.

Parameters:
<Antenna> string

Example: example "Creating emitters" on page 304

Manual operation: See "Emitter Modes" on page 181

EMITter:MODE:BEAM:OFFSet:AZIMuth <Azimuth>
EMITter:MODE:BEAM:OFFSet:ELEVation <Elevation>

Offsets the position of the beam in both the azimuth or elevation.

Parameters:
<Elevation> float
Range: -90 to 90

Example: example "Creating emitters" on page 304

Manual operation: See "Emitter Beams" on page 182

EMITter:MODE:BEAM:OFFSet:FREQuency <Frequency>

Offsets the frequency of the beam.

Parameters:
<Frequency> float
Range: -1e+09 to 1e+09

Example: example "Creating emitters" on page 304

Manual operation: See "Emitter Beams" on page 182

EMITter:MODE:BEAM:SEQuence <Sequence>

Assigns a pulse sequence, see `SEQuence:CREate`.

Parameters:
<Sequence> string

Example: example "Creating emitters" on page 304

Manual operation: See "Emitter Beams" on page 182

EMITter:MODE:BEAM:STATe <State>

Activates a beam.

Parameters:
<State> ON | OFF | 1 | 0

Example: example "Creating emitters" on page 304

Manual operation: See "Emitter Beams" on page 182

EMITter:MODE:SCAN <Scan>

Assigns an antenna scan, see `SCAN:CREate`.

Parameters:
<Scan> string

Example: example "Creating emitters" on page 304

Manual operation: See "Emitter Modes" on page 181

22.10 Generator Profiles and Instruments Commands

Example: Creating generator's profile

```
SCPI
GENerator:CATalog?
// "RS_SMW (Connected)"
GENerator:CREate "My_RS_SMW"
GENerator:LOCK
GENerator:TYPE SMW

GENerator:OPTion?
// RF:0:R&S SMW-B120|RF:1:R&S SMW-B206|BB:0:R&S SMW-B10|BB:1:R&S SMW-B10|
// DB:0:R&S SMW-B13T|SW:0:R&S SMW-K522|SW:0:R&S SMW-K512|SW:0:R&S SMW-K300|
// SW:1:R&S SMW-K300|SW:0:R&S SMW-K301|

INSTrument:COUNt?
// 4
INSTrument:CLEar
INSTrument:ADD "10.113.10.91"
INSTrument:MAP "My_RS_SMW"
INSTrument:CAPabilities?
// RFA=6e+09, BWA=1.6e+08, MEMA=1e+09, SWA=K300+K301, RFB=6e+09, BWB=1.6e+08,
// MEMB=1e+09, SWB=K300+K301
GENerator:CAPabilities?
// RFA=2e+10, BWA=1.6e+08, MEMA=1e+09, SWA=K300+K301, RFB=6e+09, BWB=1.2e+08,
// MEMB=6.4e+07, SWB=K300
INSTrument:STATus?
// 0
INSTrument:MAP "RS_SMW (Connected)"
INSTrument:STATus?
// 1
INSTrument:SELect 1
INSTrument:FIRMware?
// 3.1.18.2-3.20.012.88
```

GENerator:TYPE 307
GENerator:OPTion? 308
GENerator:LOCK 308
INSTrument:CAPabilities? 308
GENerator:CAPabilities 308
INSTrument:ADD 308
INSTrument:FIRMware? 309
INSTrument:MAP 309
INSTrument:STATus? 309

GENerator:TYPE <Type>

Sets the instrument's type.

Parameters:
<Type> SMW | SMU | SMBV | SMJ | SGT

Example: see example "Creating generator's profile" on page 307

Manual operation: See "Generator" on page 220

GENerator:OPTion?

Queries the installed options.

Return values:
<Option> string

Example: see example "Creating generator's profile" on page 307

Usage: Query only

Manual operation: See "RF Frequency Range" on page 220
See "Baseband" on page 221
See "Installed Options" on page 221

GENerator:LOCK <Lock>

Locks the created generator profile for editing exclusively per remote control. As long as the profile is locked, it cannot be edited manually in the software.

Parameters:
<Lock> ON | OFF | 1 | 0

Example: see example "Creating generator's profile" on page 307

INSTrument:CAPabilities?
GENerator:CAPabilities <Capabilities>

Queries the generator capabilities regarding supported scenario types and processing of waveforms and multi segment files.

Parameters:
<Capabilities> string

Example: see example "Creating generator's profile" on page 307

Manual operation: See "Capabilities" on page 221

INSTrument:ADD <Add>

Adds an instrument wit hthe specified IP address, computer name or the VISA resource string.

Setting parameters:
<Add> string
"<IP_Address>" or "<Computer_Name>" or "<GPIB_Address>"

Example: see example "Creating generator's profile" on page 307

Usage: Setting only

Manual operation: See "Add" on page 218

INSTrument:FIRMware?

Queries the firmware version of the selected instrument.

Return values:

<Firmware> string

Example: see example "Creating generator's profile" on page 307

Usage: Query only

Manual operation: See "List of connected instruments" on page 218

INSTrument:MAP <Map>

Maps a selected generator's profile to the selected instrument.

Use the command `INSTrument:STATus?` to query the status of the performed capabilities check.

Parameters:

<Map> string

existing generator profile
Use the command `GENerator:CATalog?` to query the existing profiles

Example: see example "Creating generator's profile" on page 307

Manual operation: See "List of connected instruments" on page 218

INSTrument:STATus?

Queries the status of the performed capabilities check.

Return values:

<Status> 0 | 1

0
status check fails: the instrument capabilities are not sufficient for the selected generator's profile
To check the capabilities, use the commands `INSTrument:CAPabilities?` and `GENerator:CAPabilities`

1
status check passed

Example: see example "Creating generator's profile" on page 307

Usage: Query only

Manual operation: See "List of connected instruments" on page 218

22.11 Inter Pulse Modulation Commands

Example: Using list and wavefrom type IPM profiles

```
SCPI
IPM:CREate "My_PRI-Stagger"
IPM:COMMent "List: 600 us, 750 us, 910 us"
IPM:UNIT SEConds
IPM:TYPE LIST
IPM:LIST:ITEM:ADD
IPM:LIST:ITEM:SELect 1
IPM:LIST:ITEM:VALue 0.0006
IPM:LIST:ITEM:REPetitions 1
IPM:LIST:ITEM:COUNt 1
IPM:LIST:ITEM:ADD
IPM:LIST:ITEM:SELect 2
IPM:LIST:ITEM:VALue 0.00075
IPM:LIST:ITEM:COUNt 2
...
IPM:LIST:SAVE "C:\_PS_files\IPM_PRI_Stagger.txt"
IPM:LIST:CLEar
// load an existing ASCII file
IPM:LIST:LOAD "C:\_PS_files\IPM_PRList.txt"

IPM:CREate "Stagger PRI"
IPM:UNIT SEConds
IPM:TYPE WAVeform
IPM:WAVeform:TYPE SINE
IPM:WAVeform:OFFSet 0.0004
IPM:WAVeform:PHASe 0
IPM:WAVeform:PKPK 0.00014
IPM:WAVeform:BASE PULSe
IPM:WAVeform:COUNt 30
```

Example: Assigning an IPM profile to a sequence

```
SCPI
SEQuence:SELect "Test Sequence"
SEQuence:ITEM:SELect 1
SEQuence:ITEM:IPM:SOURce "My_PRI-Stagger"
SEQuence:ITEM:IPM:TARGet:TYPE PARameter
SEQuence:ITEM:IPM:TARGet:PARameter PRI
SEQuence:ITEM:IPM:A 1
SEQuence:ITEM:IPM:B 0
SEQuence:ITEM:IPM:MODE INDividual
SEQuence:ITEM:IPM:RESTart 0
SEQuence:ITEM:IPM:RANDom:RESet 0
```

Example: Using the step IPM profile

```
SCPI
IPM:CREate "FHOP"
IPM:COMMent "Frequency Hops"
IPM:UNIT HERTz
IPM:TYPE STEPs
IPM:STEP:STARt 0
IPM:STEP:INCRement 1e+06
IPM:STEP:STEPs 5
IPM:STEP:BURSt 2
```

Example: Using PRI pofiles of random type

```
SCPI
IPM:CREate "My_PRI-Jitter"
IPM:UNIT SEConds
IPM:TYPE RANDom
IPM:RANDom:DISTribution UNIForm
IPM:RANDom:UNIForm:MINimum 0.0011
IPM:RANDom:UNIForm:MAXimum 0.00115
IPM:RANDom:UNIForm:STEP 2.5e-06

IPM:CREate "Test"
IPM:UNIT SEConds
IPM:TYPE RSTEp
IPM:RSTep:MINimum 0.0005
IPM:RSTep:MAXimum 0.005
IPM:RSTep:STEP:MINimum 0.0001
IPM:RSTep:STEP:MAXimum 0.0005
IPM:RSTep:PERiod 10

IPM:CREate "Test2"
IPM:UNIT HERTz
IPM:TYPE RLISt
// create a List (see the "My_PRI-Stagger" example)
IPM:RLISt:BURSt 3
IPM:RLISt:REUSe 0
```

Example: Using the interpolated shape IPM profile

```
SCPI
IPM:CREate "Custom IPM"
IPM:UNIT DB
IPM:TYPE SHAPe
// create a List (see the "My_PRI-Stagger" example)
IPM:SHAPe:BASE PULSE
IPM:SHAPe:COUNt 10
// IPM:SHAPe:BASE TIME
// IPM:SHAPe:PERiod 0.5
IPM:SHAPe:INTerpol NONE
```

IPM:TYPE ... 313
IPM:UNIT ... 313
SEQuence:ITEM:IPM:TARGet:PARameter ... 313
SEQuence:ITEM:IPM:TARGet:TYPE ... 313
SEQuence:ITEM:IPM:TARGet:VARiable ... 314
IPM:EQUation ... 314
IPM:LIST:ITEM:VALue ... 314
IPM:LIST:ITEM:REPetition ... 314
IPM:LIST:LOAD ... 314
IPM:LIST:SAVE ... 315
IPM:RANDom:DISTribution ... 315
IPM:RANDom:NORMal:LIMit ... 315
IPM:RANDom:NORMal:MEAN ... 315
IPM:RANDom:NORMal:STD ... 316
IPM:RANDom:U:CENTer ... 316
IPM:RANDom:U:RANGe ... 316
IPM:RANDom:UNIForm:MINimum ... 316
IPM:RANDom:UNIForm:MAXimum ... 316
IPM:RANDom:UNIForm:STEP ... 317
IPM:RLISt:BURSt ... 317
IPM:RLISt:REUSe ... 317
IPM:SHAPe:BASE ... 317
IPM:SHAPe:COUNt ... 317
IPM:SHAPe:INTerpol ... 318
IPM:SHAPe:PERiod ... 318
IPM:STEP:BURSt ... 318
IPM:STEP:INCRement ... 318
IPM:STEP:STARt ... 319
IPM:STEP:STEPs ... 319
IPM:RSTep:MINimum ... 319
IPM:RSTep:MAXimum ... 319
IPM:RSTep:STEP:MINimum ... 319
IPM:RSTep:STEP:MAXimum ... 319
IPM:RSTep:PERiod ... 319
IPM:WAVeform:TYPE ... 320
IPM:WAVeform:BASE ... 320
IPM:WAVeform:COUNt ... 320
IPM:WAVeform:OFFSet ... 320

IPM:WAVeform:PHASe 321
IPM:WAVeform:PERiod 321
IPM:WAVeform:PKPK 321
SEQuence:ITEM:IPM:A 321
SEQuence:ITEM:IPM:B 321
SEQuence:ITEM:IPM:MODE 322
SEQuence:ITEM:IPM:RANDom:RESet 322
SEQuence:ITEM:IPM:RESTart 322
SEQuence:ITEM:IPM:SOURce 322

IPM:TYPE <Type>

Sets the shape of the profile.

Parameters:

<Type> STEPs | WAVeform | RLISt | LIST | SHAPe | RANDom | EQUation | PLUGin | RSTep

Example: see Using list and wavefrom type IPM profiles

Manual operation: See "Profile" on page 131

IPM:UNIT <Unit>

Sets the units of the profile.

Parameters:

<Unit> NONE | SEConds | HERTz | DB | DEGRees | PERCent

Example: see Using list and wavefrom type IPM profiles

Manual operation: See "Unit of Affected Parameter" on page 131

SEQuence:ITEM:IPM:TARGet:PARameter <Parameter>

Sets the pulse parameter the IPM profile is assigned to.

Parameters:

<Parameter> OVERshoot | LEVel | ATOP | RLEVel | DROop | AMDepth | RISE | WIDTh | DELay | FMDeviation | FALL | FMFRequency | CDEViation | ABASe | RFRequency | PRF | PHASe | AMFRequency | PRI | FREQuency

Example: see Assigning an IPM profile to a sequence

Manual operation: See "Assign to" on page 143

SEQuence:ITEM:IPM:TARGet:TYPE <Type>

Sets whether the profile is assigned to a parameter or to a variable.

Parameters:

<Type> PARameter | VARiable

Example: see Assigning an IPM profile to a sequence

Manual operation: See "Assign to" on page 143

SEQuence:ITEM:IPM:TARGet:VARiable <Variable>

Sets the variable the profile is assigned to.

Parameters:
<Variable> string

Manual operation: See "Assign to" on page 143

IPM:EQUation <Equation>

Defines the IPM shape as a function.

Parameters:
<Equation> string

Example:
```
IPM:EQUation "(i<10) ? 5 : (i*0.5)"
```

Manual operation: See "Equation" on page 136

IPM:LIST:ITEM:VALue <Value>

Sets the value of the selected list item.

Parameters:
<Value> float
Range: -1e+11 to 1e+11

Example: see example "Using list and wavefrom type IPM profiles" on page 310

Manual operation: See "List" on page 133

IPM:LIST:ITEM:REPetition <Repetition>

Sets the number of times a list item is repeated.

Parameters:
<Repetition> float
Range: 1 to 1e+09

Example: see Using list and wavefrom type IPM profiles

Manual operation: See "List" on page 133

IPM:LIST:LOAD <Load>

Loads an IPM profile form an ASCII file.

Parameters:
<Load> string
file path, file name and file sxtension

Example: see Using list and wavefrom type IPM profiles

Manual operation: See "Edit List" on page 140

IPM:LIST:SAVE <Save>

Stores the IPM profile as a file.

Parameters:
<Save> string
File path incl. file name and extension.

Example: see Using list and wavefrom type IPM profiles

Manual operation: See "Edit List" on page 140

IPM:RANDom:DISTribution <Distribution>

Sets the distribution function.

Parameters:
<Distribution> UNIForm | NORMal | U

Example: see example "Using PRI pofiles of random type" on page 311

Manual operation: See "Random" on page 138

IPM:RANDom:NORMal:LIMit <Limit>

Sets the limit parameter of the normal distribution function.

Parameters:
<Limit> float
Range: -1e+09 to 1e+09

Example: see `IPM:RANDom:NORMal:MEAN` on page 315

Manual operation: See "Random" on page 138

IPM:RANDom:NORMal:MEAN <Mean>

Sets the mean parameter of the normal distribution function.

Parameters:
<Mean> float
Range: -1e+09 to 1e+09

Example:
```
IPM:RANDom:DISTribution NORMal
IPM:RANDom:NORMal:MEAN 0
IPM:RANDom:NORMal:STD 1
IPM:RANDom:NORMal:LIMit 3
```

Manual operation: See "Random" on page 138

IPM:RANDom:NORMal:STD <Std>

Sets the standard deviation parameter of the normal distribution function.

Parameters:
<Std> float
Range: 1e-09 to 1e+06

Example: see `IPM:RANDom:NORMal:MEAN` on page 315

Manual operation: See "Random" on page 138

IPM:RANDom:U:CENTer <Center>

Sets the center parameter of the U distribution.

Parameters:
<Center> float
Range: -1e+09 to 1e+09

Manual operation: See "Random" on page 138

IPM:RANDom:U:RANGe <Range>

Sets the range parameter of the U distribution.

Parameters:
<Range> float
Range: 1e-09 to 1e+09

Manual operation: See "Random" on page 138

IPM:RANDom:UNIForm:MINimum <Minimum>
IPM:RANDom:UNIForm:MAXimum <Maximum>

Sets the range of the uniform distribution function.

Parameters:
<Maximum> float
Range: -1e+09 to 1e+09

Example: see example "Using PRI pofiles of random type" on page 311

Manual operation: See "Random" on page 138

IPM:RANDom:UNIForm:STEP <Step>

Sets the granularity of the uniform distribution function.

Parameters:
<Step> float
Range: 1e-09 to 1e+09

Example: see example "Using PRI pofiles of random type" on page 311

Manual operation: See "Random" on page 138

IPM:RLISt:BURSt <Burst>

Defines how many time an increment is repeated.

Parameters:
<Burst> float
Range: 1 to 1000

Example:

Manual operation: See "Random List" on page 136

IPM:RLISt:REUSe <Reuse>

If disabled, each value is used only once.

Parameters:
<Reuse> ON | OFF | 1 | 0

Example:
```
IPM:RLISt:REUSe 0
```

Manual operation: See "Random List" on page 136

IPM:SHAPe:BASE <Base>

Defines the way the list items are processed.

Parameters:
<Base> PULSe | TIME

Example: see Using the interpolated shape IPM profile

Manual operation: See "Interpolated Shape" on page 135

IPM:SHAPe:COUNt <Count>

Sets the number of pulses for that the data from the list is used.

Parameters:
<Count> integer
Range: 1 to 1e+09

Example: see Using the interpolated shape IPM profile

Manual operation: See "Interpolated Shape" on page 135

IPM:SHAPe:INTerpol <Interpol>

Enables a linear transition between the increments.

Parameters:
<Interpol> LINear | NONE

Example: see Using the interpolated shape IPM profile

Manual operation: See "Interpolated Shape" on page 135

IPM:SHAPe:PERiod <Period>

Sets the period of time over that the list items are equally distributed.

Parameters:
<Period> float
Range: 1e-09 to 1e+09

Example: see Using the interpolated shape IPM profile

Manual operation: See "Interpolated Shape" on page 135

IPM:STEP:BURSt <Burst>

Sets the number of times an increment is repeated.

Parameters:
<Burst> float
Range: 1 to 1000

Example: see example "Using the step IPM profile" on page 311

Manual operation: See "Steps" on page 132

IPM:STEP:INCRement <Increment>

Sets the step size.

Parameters:
<Increment> float
Range: -1e+09 to 1e+09

Example: see example "Using the step IPM profile" on page 311

Manual operation: See "Steps" on page 132

IPM:STEP:STARt <Start>

Sets the start value.

Parameters:
<Start> float
Range: -1e+09 to 1e+09

Example: see example "Using the step IPM profile" on page 311

Manual operation: See "Steps" on page 132

IPM:STEP:STEPs <Steps>

Sets the number of steps.

Parameters:
<Steps> float
Range: 1 to 10000

Example: see example "Using the step IPM profile" on page 311

Manual operation: See "Steps" on page 132

IPM:RSTep:MINimum <Minimum>
IPM:RSTep:MAXimum <Maximum>

Sets the value range.

Parameters:
<Maximum> float
Range: 0 to 1e+11

Example: see example "Using PRI pofiles of random type" on page 311

Manual operation: See "Random Steps" on page 137

IPM:RSTep:STEP:MINimum <Minimum>
IPM:RSTep:STEP:MAXimum <Maximum>

Sets the step size range.

Parameters:
<Maximum> float
Range: 0.1 to 0.5

Example: see example "Using PRI pofiles of random type" on page 311

Manual operation: See "Random Steps" on page 137

IPM:RSTep:PERiod <Period>

Sets the pattern length.

Parameters:
<Period> float
Range: 0 to 4096

Example: see example "Using PRI pofiles of random type" on page 311

Manual operation: See "Random Steps" on page 137

IPM:WAVeform:TYPE <Type>

Sets the profile shape.

Parameters:
<Type> RAMP | SINE | TRIangular

Example: see Using list and wavefrom type IPM profiles

Manual operation: See "Waveform" on page 134

IPM:WAVeform:BASE <Base>

Defines how the waveform period is defined, as a time duration or as a number of pulses.

Parameters:
<Base> PULSe | TIME

Example: see Using list and wavefrom type IPM profiles

Manual operation: See "Waveform" on page 134

IPM:WAVeform:COUNt <Count>

Sets the waveform period as number of pulses.

Parameters:
<Count> integer
Range: 1 to 1e+09

Example: see Using list and wavefrom type IPM profiles

Manual operation: See "Waveform" on page 134

IPM:WAVeform:OFFSet <Offset>

Shifts the profile by the selected offset.

Parameters:
<Offset> float
Range: -1e+09 to 1e+09

Example: see Using list and wavefrom type IPM profiles

Manual operation: See "Waveform" on page 134

IPM:WAVeform:PHASe <Phase>

Enables a phase offset to change the start phase of the sine wave.

Parameters:

<Phase> float
Range: -1e+09 to 1e+09
Default unit: sec

Example: see Using list and wavefrom type IPM profiles

Manual operation: See "Waveform" on page 134

IPM:WAVeform:PERiod <Period>

Sets the waveform period.

Parameters:

<Period> float
Range: 1e-09 to 1e+09
Default unit: sec

Example: see Using list and wavefrom type IPM profiles

Manual operation: See "Waveform" on page 134

IPM:WAVeform:PKPK <Pkpk>

Sets the value range of the linear ramp profile or the period of the sine profile.

Parameters:

<Pkpk> float
Range: 1e-09 to 1e+09
Default unit: sec

Example: see Using list and wavefrom type IPM profiles

Manual operation: See "Waveform" on page 134

SEQuence:ITEM:IPM:A <A>
SEQuence:ITEM:IPM:B <B>

Sets the paramters `a` and `b` of the transformation `y = a*x+b`.

Parameters:

<B> float
Range: -1e+09 to 1e+09

Example: see Assigning an IPM profile to a sequence

Manual operation: See "Transformation" on page 143

SEQuence:ITEM:IPM:MODE <Mode>

Defines the way the variations are applied on repeating pulses.

Parameters:
<Mode> INDividual | SAME

Example: see Assigning an IPM profile to a sequence

Manual operation: See "Repetition" on page 143

SEQuence:ITEM:IPM:RANDom:RESet <Reset>

Resets the start seed of random generator.

Parameters:
<Reset> ON | OFF | 1 | 0

Example: see Assigning an IPM profile to a sequence

Manual operation: See "Restart" on page 144

SEQuence:ITEM:IPM:RESTart <Restart>

Restarts the IPM for this sequence line item.

Parameters:
<Restart> ON | OFF | 1 | 0

Example: see Assigning an IPM profile to a sequence

Manual operation: See "Restart" on page 144

SEQuence:ITEM:IPM:SOURce <Source>

Selects the profile source.

Use the command `IPM:CATalog?` to querry the exsisting prfiles.

Parameters:
<Source> string

Example: see Assigning an IPM profile to a sequence

Manual operation: See "Source Profile" on page 143

22.12 Marker Commands

Marker commands enable or up to four markers in any combination at once.

Markers signals are binary signals. Each marker signal is represented by a single bit within a marker byte. The table 22-1 explains the assailment.

Table 22-1: Setting parameter as function of the marker states

Marker	Marker byte	Value of the settings parameter in the remote commands
M1	0000 0001	0x01
M2	0000 0010	0x02
M3	0000 0100	0x04
M4	0000 1000	0x08

In the following examples, we assume that a pulse-base sequence "Test Sequence" and the required Pulses have been created.

See also:

- example "Handling items" on page 272
- example "Creating an unmodulated pulse" on page 331

Example: Defining pulse and sequence markers and enabling the global markers

```
SCPI
PULSe:SELect "P1"
// activate Marker 1 for the whole pulse on time (rise, fall, with time)
PULSe:MARKer:RISE 1
PULSe:MARKer:FALL 1
PULSe:MARKer:WIDTh 1
// enable Marker 2 on restart
PULSe:MARKer:AUTO 2
// enable Marker 3 als gate marker
PULSe:MARKer:GATE 4

SEQuence:SELect "Test Sequence"
SEQuence:ITEM:SELect 1
SEQuence:ITEM:MARKer:FIRSt 2
SEQuence:ITEM:MARKer:LAST 4
SEQuence:ITEM:MARKer:ALL 1

SCENario:OUTPut:MARKer:ENABle 1
SCENario:OUTPut:MARKer:SCENario:ENABle 0
// SCENario:OUTPut:MARKer:SCENario:DURation 0.01
```

PULSe:MARKer:RISE 324
PULSe:MARKer:WIDTh 324
PULSe:MARKer:FALL 324
PULSe:MARKer:AUTO 324
PULSe:MARKer:GATE 324
SEQuence:ITEM:MARKer:FIRSt 324
SEQuence:ITEM:MARKer:LAST 324
SEQuence:ITEM:MARKer:ALL 324
SCENario:OUTPut:MARKer:ENABle 325
SCENario:OUTPut:MARKer:SCENario:ENABle 325
SCENario:OUTPut:MARKer:SCENario:DURation 325

PULSe:MARKer:RISE <Rise>
PULSe:MARKer:WIDTh <Width>
PULSe:MARKer:FALL <Fall>

Enables up to four markers of the corresponding type.

Parameters:

<Fall> float
see table 22-1.
Range: 0 to 65535

Example: see example "Defining pulse and sequence markers and enabling the global markers" on page 323

Manual operation: See "Pulse markers" on page 236

PULSe:MARKer:AUTO <Auto>

Enables up to four restart markers.

Parameters:

<Auto> float
see table 22-1.
Range: 0 to 65535

Example: see example "Defining pulse and sequence markers and enabling the global markers" on page 323

Manual operation: See "Pulse markers" on page 236

PULSe:MARKer:GATE <Gate>

Enables up to four gate markers.

Parameters:

<Gate> float
see table 22-1.
Range: 0 to 65535

Example: see example "Defining pulse and sequence markers and enabling the global markers" on page 323

Manual operation: See "Pulse markers" on page 236

SEQuence:ITEM:MARKer:FIRSt <First>
SEQuence:ITEM:MARKer:LAST <Last>
SEQuence:ITEM:MARKer:ALL <All>

Enables up to four markers of the corresponding type.

Parameters:

<All> float

see table 22-1.

Range: 0 to 65535

Example: see example "Defining pulse and sequence markers and enabling the global markers" on page 323

Manual operation: See "Sequence markers" on page 237

SCENario:OUTPut:MARKer:ENABle <Enable>

Enables that markers are considered by the generation of the output waveform file.

Parameters:

<Enable> ON | OFF | 1 | 0

Example: see example "Defining pulse and sequence markers and enabling the global markers" on page 323

Manual operation: See "Features" on page 249

SCENario:OUTPut:MARKer:SCENario:ENABle <Enable>

Enables an additional marker, that is held high from the scenario start until the duration, selected with the command `SCENario:OUTPut:MARKer:SCENario:DURation`.

Parameters:

<Enable> ON | OFF | 1 | 0

Example: see example "Defining pulse and sequence markers and enabling the global markers" on page 323

Manual operation: See "Features" on page 249

SCENario:OUTPut:MARKer:SCENario:DURation <Duration>

Sets the duration of the scenario marker.

Parameters:

<Duration> float

Range: 0 to 1
Increment: 1 ms
Default unit: sec

Example: see example "Defining pulse and sequence markers and enabling the global markers" on page 323

Manual operation: See "Features" on page 249

22.13 Plugin and Reporting Commands

Example: Laoding plugins in the repository

```
PLUGin:CREate "My_QAM_PlugIn"
PLUGin:LOAD "C:\_PS_files\QAM.dll"
PLUGin:MODule:AUTHor?
// Rohde&Schwarz
PLUGin:MODule:NAME?
// QAM
PLUGin:MODule:TYPE?
// MOP
PLUGin:MODule:VERSion?
// 2.0.0
PLUGin:MODule:COMMent?
// This plugin creates a QAM modulation.
// Parameters:
// Rate [Hz]     symbol rate
PLUGin:MODule:DATA?
// 0
```

Example: Using a plugin as a modulaiton source

```
SCPI
PULSe:CREate "Custom QAM"
PULSe:MOP:ENABle 1
PULSe:MOP:TYPE PLUGin
PLUGin:CATalog?
// "My_QAM_PlugIn","CustomIPM"
PULSe:MOP:PLUGin:NAME "My_QAM_PlugIn"
PULSe:MOP:PLUGin:VARiable:CATalog?
// "Rate","Type"
PULSe:MOP:PLUGin:VARiable:SELect "Rate"
PULSe:MOP:PLUGin:VARiable:VALue?
// 100000
PULSe:MOP:PLUGin:VARiable:SELect "Type"
PULSe:MOP:PLUGin:VARiable:VALue?
// 2
// selected is the third value from the list of variables
DSRC:CATalog?
// "DataSource"
PULSe:MOP:DATA:DSRC "DataSource"
PULSe:MOP:DATA:CODing NONE
PULSe:MOP:FILTer:TYPE COS
PULSe:MOP:FILTer:LENGth 128
PULSe:MOP:FILTer:BT 0.3
PULSe:MOP:FILTer:BWIDth 1000
```

Example: Using a plugin as an IPM profile

```
SCPI
IPM:CREate "Custom IPM"
IPM:UNIT PERCent
IPM:TYPE PLUGin
PLUGin:CATalog?
// "My_QAM_PlugIn","CustomIPM"
IPM:PLUGin:NAME "CustomIPM"
IPM:PLUGin:VARiable:CATalog?
// "$step","$pw"
IPM:PLUGin:VARiable:SELect "$pw"
IPM:PLUGin:VARiable:VALue 0.00001
```

Example: Generating reports

```
SCPI
SCENario:PDW:ENABle 1
SCENario:PDW:PATH "C:\_My_Report_Files"
SCENario:PDW:TYPE PLUGin
PLUGin:CATalog?
// "My_QAM_PlugIn","CustomIPM","Report PDW","Reporting DFS"
SCENario:PDW:PLUGin:NAME "Report PDW"
SCENario:PDW:PLUGin:VARiable:CATalog?
// "File","Header"
SCENario:PDW:PLUGin:VARiable:SELect "File"
SCENario:PDW:PLUGin:VARiable:VALue?
// "pdw_out.txt"
```

PLUGin:LOAD 328
PLUGin:MODule:AUTHor? 328
PLUGin:MODule:COMMent? 328
PLUGin:MODule:VERSion? 328
PLUGin:MODule:TYPE? 328
PLUGin:MODule:DATA? 328
SCENario:PDW:PLUGin:VARiable:CATalog? 329
IPM:PLUGin:VARiable:CATalog? 329
PULSe:MOP:PLUGin:VARiable:CATalog? 329
SCENario:PDW:PLUGin:VARiable:SELect 329
IPM:PLUGin:VARiable:SELect 329
PULSe:MOP:PLUGin:VARiable:SELect 329
SCENario:PDW:PLUGin:VARiable:VALue 329
IPM:PLUGin:VARiable:VALue 329
PULSe:MOP:PLUGin:VARiable:VALue 329
SCENario:PDW:ENABle 329
SCENario:PDW:PATH 329
SCENario:PDW:TYPE 330
SCENario:PDW:PLUGin:NAME 330
SCENario:PDW:TEMPlate 330

PLUGin:LOAD <Load>

Loads the selected DLL file, see also chapter A.3, "Plugin Programming API", on page 400.

Setting parameters:

<Load> string

file path incl. file name and extension

Example: see example "Laoding plugins in the repository" on page 326

Usage: Setting only

PLUGin:MODule:AUTHor?
PLUGin:MODule:COMMent?
PLUGin:MODule:VERSion?
PLUGin:MODule:TYPE?

Queries information on the loaded file.

The query returns information as specified in the description of the corresponding function in chapter A.3, "Plugin Programming API", on page 400.

The following is are the possible values for the type query.

Return values:

<Type> MOP | REPort | IPM

MOP
Plugin for MOP

IPM
Plugin for IPM

REPort
Plugin for reports created during the waveform generation

Example: see example "Laoding plugins in the repository" on page 326

Usage: Query only

PLUGin:MODule:DATA?

Queries whether the plugin requires data from a data source.

Return values:

<Data> 0 | 1

0
data source is not required

1
data source is required

*RST: 0

Example: see example "Laoding plugins in the repository" on page 326

Usage: Query only

SCENario:PDW:PLUGin:VARiable:CATalog?
IPM:PLUGin:VARiable:CATalog?
PULSe:MOP:PLUGin:VARiable:CATalog?

Queries the variables used in the plugin.

Return values:
<Catalog> string

Example: see Using a plugin as a modulaiton source

Usage: Query only

Manual operation: See "Plugin Variables" on page 94

SCENario:PDW:PLUGin:VARiable:SELect <Select>
IPM:PLUGin:VARiable:SELect <Select>
PULSe:MOP:PLUGin:VARiable:SELect <Select>

Selects a plugin variable.

Parameters:
<Select> string

Example: see Using a plugin as a modulaiton source

Manual operation: See "Plugin Variables" on page 94

SCENario:PDW:PLUGin:VARiable:VALue <Value>
IPM:PLUGin:VARiable:VALue <Value>
PULSe:MOP:PLUGin:VARiable:VALue <Value>

Sets the values of the selected variable.

Parameters:
<Value> string

Example: see Using a plugin as a modulaiton source

Manual operation: See "Plugin Variables" on page 94

SCENario:PDW:ENABle <Enable>

Enabels generation of Pulse Descripter Word (PDW) reports.

Parameters:
<Enable> ON | OFF | 1 | 0

Example: see example "Generating reports" on page 327

Manual operation: See "Reporting Enable" on page 262

SCENario:PDW:PATH <Path>

Sets the target directory in that the generated report files are stored.

Parameters:
<Path> string

Example: see example "Generating reports" on page 327

Manual operation: See "Target Path, Set Path" on page 262

SCENario:PDW:TYPE <Type>

Sets the template used be the reporting function.

Parameters:
<Type> DEFault | TEMPlate | PLUGin

Manual operation: See "Type" on page 262

SCENario:PDW:PLUGin:NAME <Name>

Selects and loads a reporting template. This template must exist in the "Plugin" library.

To query a list of availabel plugins, use the command `PLUGin:CATalog?`.

Parameters:
<Name> string

Example: see example "Generating reports" on page 327

Manual operation: See "Plugin and Plugin Variables" on page 263

SCENario:PDW:TEMPlate <Template>

Edits the selected template.

Parameters:
<Template> string

Example:

```
SCPI
SCENario:PDW:TEMPlate?
# Pulse Sequencer PDW Report
#
Date:       <ISODATE>
Repository: <REPOSITORY>
Scenario:   <SCENARIO>

          TOA |        RF |        PW |      PA | MF |  MOP |      BW |
           ns |       GHz |        us |     dBm |    |      |     kHz |
-------------------------------------------------------------------------
.HDR
 <TOA> | <RF> | <PW> | <PA> |  <MF> | <MOP> | <BW> |
# END OF REPORT
.OPT <TOA:12.0f:-9>
.OPT <RF:9.6f:9>
.OPT <PW:9.3f:-6>
.OPT <PA:6.1f:0>
.OPT <BW:6.0f:3>
```

Manual operation: See "Edit Template" on page 262

22.14 Pulse Commands

The following are examples on how to create and configure pulses in remote environment.

Example: Creating an unmodulated pulse

```
SCPI
PULSe:CREate "P_1"
PULSe:CATalog?
// "P_1"
PULSe:SELect "P_1"
PULSe:COMMent "PW=100us"
PULSe:TIME:REFerence FULL
PULse:TIME:WIDTh 100 us
PULSe:TIME:RISE 10 us
PULSe:TIME:FALL 10 us
PULSe:TYPE:RISE LINear
PULSe:TYPE:FALL LINear
PULSe:MOP:ENABle 0
PULSe:MOP:EXCLude:ENABle 0
// remane the pulse
PULSe:NAME "P1"
```

Example: Creating a linear chirp pulse

```
SCPI
PULSe:CREate "LinearChirp"
PULSe:CUSTom 0
PULSe:TIME:REFerence POWer
PULSe:TIME:RISE 1e-05
PULSe:TYPE:RISE RCOSine
PULSe:TIME:WIDTh 0.0001
PULSe:TIME:FALL 1e-05
PULSe:TYPE:FALL RCOSine
PULSe:LEVel:ON 0
PULSe:LEVel:OFF 100
PULSe:LEVel:DROop 1
PULSe:RIPPle 0
PULSe:RIPPle:FREQuency 50000

PULSe:MOP:ENABle 1
PULSe:MOP:TYPE CHIRP
PULSe:MOP:CHIRp:TYPE UP
PULSe:MOP:CHIRp:DEViation 5e+06
PULSe:MOP:EXCLude:MODE LEVel
PULSe:MOP:EXCLude:LEVel:STARt 25
PULSe:MOP:EXCLude:LEVel:STOP 25
```

Example: Creating a FM step pulse

```
SCPI
PULSe:CREate "FM_Step"
PULSe:TIME:REFerence FULL
PULSe:TIME:RISE 0
PULSe:TIME:WIDTh 0.000125
PULSe:TIME:FALL 0
PULSe:LEVel:ON 0
PULSe:LEVel:OFF 100
PULSe:MOP:EXCLude:MODE WIDTh

PULSe:MOP:ENABle 1
PULSe:MOP:COMMent "fstart = -50 MHz, fend = 50 MHz, Df = 25 MHz (N = 5)"
PULSe:MOP:TYPE FMSTep
PULSe:MOP:FMSTep:ADD
PULSe:MOP:FMSTep:SELect 1
PULSe:MOP:FMSTep:DURation 2.49999993684469e-05
PULSe:MOP:FMSTep:FREQuency -50000000
PULSe:MOP:FMSTep:ADD
PULSe:MOP:FMSTep:SELect 2
PULSe:MOP:FMSTep:DURation 2.49999993684469e-05
PULSe:MOP:FMSTep:FREQuency -25000000
...
```

Example: Creating a BPSK pulse

```
SCPI
PULSe:CREate "Test"
PULSe:MOP:ENABle 1
PULSe:MOP:TYPE BPSK
PULSe:MOP:BPSK:TYPE NORMal
PULSe:MOP:BPSK:SRATe:AUTO 0
PULSe:MOP:BPSK:SRATe 1e+06
PULSe:MOP:BPSK:PHASe 180
PULSe:MOP:BPSK:TTYPe COSine
PULSe:MOP:BPSK:TTIMe 5
```

Example: Creating pulses with custom envelope

```
SCPI
PULSe:SELect "Test"
PULSe:CUSTom 1
PULSe:ENVelope:MODE DATA
PULSe:ENVelope:DATA:LOAD "C:\_PS_files\mop shape custom.csv"
PULSe:ENVelope:DATA:UNIT VOLTAGE
PULSe:ENVelope:DATA:MULTiplier 1
PULSe:ENVelope:DATA:OFFSet 0
PULSe:ENVelope:DATA:ITEM:SELect 1
PULSe:ENVelope:DATA:ITEM:VALue?
// 0.2
PULSe:ENVelope:DATA:ITEM:SELect 7
PULSe:ENVelope:DATA:ITEM:VALue?
// 0.3
PULSe:ENVelope:DATA:ITEM:COUNt?
// 7

// PULSe:ENVelope:MODE EQUation
// PULSe:ENVelope:EQUation "# Sech Pulse
// 1 / ( cosh( ( 2 * (t-Tr-Ton/2) * ln( 1+sqrt(2) ) ) / Ton ) )^2"
```

PULSe:CUSTom....335
PULSe:ENVelope:DATA:ITEM:VALue....335
PULSe:ENVelope:DATA:LOAD....335
PULSe:ENVelope:DATA:UNIT....335
PULSe:ENVelope:DATA:MULTiplier....336
PULSe:ENVelope:DATA:OFFSet....336
PULSe:ENVelope:DATA:SAVE....336
PULSe:ENVelope:EQUation....336
PULSe:ENVelope:MODE....337
PULSe:LEVel:DROop....337
PULSe:LEVel:OFF....337
PULSe:LEVel:ON....337
PULSe:MOP:AM:FREQuency....337
PULSe:MOP:AM:MDEPth....337
PULSe:MOP:AM:TYPE....338

PULSe:MOP:AMSTep:DURation ... 338
PULSe:MOP:FMSTep:DURation ... 338
PULSe:MOP:AMSTep:LEVel ... 338
PULSe:MOP:FMSTep:FREQuency ... 338
PULSe:MOP:ASK:INVert ... 339
PULSe:MOP:ASK:MDEPth ... 339
PULSe:MOP:ASK:SRATe ... 339
PULSe:MOP:BARKer:CODE ... 339
PULSe:MOP:BARKer:TTIMe ... 340
PULSe:MOP:BPSK:PHASe ... 340
PULSe:MOP:BPSK:SRATe ... 340
PULSe:MOP:BPSK:SRATe:AUTO ... 340
PULSe:MOP:BPSK:TTIMe ... 341
PULSe:MOP:BPSK:TTYPe ... 341
PULSe:MOP:BPSK:TYPE ... 341
PULSe:MOP:CHIRp:DEViation ... 341
PULSe:MOP:CHIRp:TYPE ... 341
PULSe:MOP:DATA:CODing ... 342
PULSe:MOP:DATA:DSRC ... 342
PULSe:MOP:ENABle ... 342
PULSe:MOP:EXCLude:ENABle ... 342
PULSe:MOP:EXCLude:LEVel:STARt ... 342
PULSe:MOP:EXCLude:LEVel:STOP ... 342
PULSe:MOP:EXCLude:MODE ... 343
PULSe:MOP:EXCLude:TIME:STARt ... 343
PULSe:MOP:EXCLude:TIME:STOP ... 343
PULSe:MOP:FILTer:BT ... 343
PULSe:MOP:FILTer:BWIDth ... 343
PULSe:MOP:FILTer:LENGth ... 344
PULSe:MOP:FILTer:ROLLoff ... 344
PULSe:MOP:FILTer:TYPE ... 344
PULSe:MOP:FM:DEViation ... 344
PULSe:MOP:FM:FREQuency ... 345
PULSe:MOP:FSK:DEViation ... 345
PULSe:MOP:FSK:SRATe ... 345
PULSe:MOP:NLCHirp:EQUation ... 345
PULSe:MOP:NOISe:BWIDth ... 345
PULSe:MOP:PCHirp:COEFficient ... 346
PULSe:MOP:PCHirp:TERM ... 346
PULSe:MOP:PLISt:VALue ... 346
PULSe:MOP:POLY:LENGth ... 346
PULSe:MOP:POLY:TYPE ... 346
PULSe:MOP:QPSK:SRATe ... 347
PULSe:MOP:QPSK:TYPE ... 347
PULSe:MOP:TYPE ... 347
PULSe:OVERshoot ... 347
PULSe:OVERshoot:DECay ... 347
PULSe:RIPPle ... 348
PULSe:RIPPle:FREQuency ... 348
PULSe:TIME:FALL ... 348

PULSe:TIME:RISE....348
PULSe:TIME:REFerence....348
PULSe:TIME:WIDTh....349
PULSe:TYPE:FALL....349
PULSe:TYPE:RISE....349
PULSe:SETTings....349
PULSe:PREView:MODE....349
PULSe:PREView:MOP....350

PULSe:CUSTom <Custom>

Enables the use of a custom envelope function

Parameters:
<Custom> ON | OFF | 1 | 0

Manual operation: See "Custom Envelope" on page 70

PULSe:ENVelope:DATA:ITEM:VALue <Value>

Sets the value of the selected item.

Parameters:
<Value> float
Range: -1e+09 to 1e+09

Example: see example "Creating pulses with custom envelope" on page 333

Manual operation: See "Envelope Definition as a Function form Imported Data" on page 77

PULSe:ENVelope:DATA:LOAD <Load>

Loads an envelope description form an ASCII file.

Parameters:
<Load> string
file path, file name and file extension

Example: see example "Creating pulses with custom envelope" on page 333

Manual operation: See "Envelope Definition as a Function form Imported Data" on page 77

PULSe:ENVelope:DATA:UNIT <Unit>

Sets the data format.

Parameters:
<Unit> VOLTage | WATTs | DB

Example: see example "Creating pulses with custom envelope" on page 333

Manual operation: See "Envelope Definition as a Function form Imported Data" on page 77

PULSe:ENVelope:DATA:MULTiplier <Multiplier>

Sets a multiplier factor.

Parameters:

<Multiplier> float

Range: -100 to 100

Example: see example "Creating pulses with custom envelope" on page 333

Manual operation: See "Envelope Definition as a Function form Imported Data" on page 77

PULSe:ENVelope:DATA:OFFSet <Offset>

Sets an offset for the envelope.

Parameters:

<Offset> float

Range: -100 to 100

Example: see example "Creating pulses with custom envelope" on page 333

Manual operation: See "Envelope Definition as a Function form Imported Data" on page 77

PULSe:ENVelope:DATA:SAVE <Save>

Stores the custom envelpe into file.

Parameters:

<Save> string

file path, file name and file extension

Example:

```
PULSe:ENVelope:DATA:LOAD "C:
\_PS_files\mop_shape.csv"
```

PULSe:ENVelope:EQUation <Equation>

Determines the envelope mathematically.

Parameters:

<Equation> string

Manual operation: See "Envelope Definition as an Equation" on page 76

PULSe:ENVelope:MODE <Mode>

Selects the type of the custom envelope function.

Parameters:
<Mode> DATA | EQUation

Example: see example "Creating pulses with custom envelope" on page 333

Manual operation: See "Custom Envelope" on page 70

PULSe:LEVel:DROop <Droop>

Sets the amplitude droop.

Parameters:
<Droop> float
Range: 0 to 50

Example: see example "Creating a linear chirp pulse" on page 332

Manual operation: See "Attenuation" on page 73

PULSe:LEVel:OFF <Off>
PULSe:LEVel:ON <On>

Sets the power during the pulse on time or the pulse off time.

Parameters:
<On> float
Range: 0 to 100

Example: see example "Creating a linear chirp pulse" on page 332

Manual operation: See "Attenuation" on page 73

PULSe:MOP:AM:FREQuency <Frequency>

Sets modulation frequency.

Parameters:
<Frequency> float
Range: 0.001 to 1e+09

Example: see `PULSe:MOP:AM:TYPE` on page 338

Manual operation: See "AM" on page 83

PULSe:MOP:AM:MDEPth <Mdepth>

Sets the modulation depth.

Parameters:
<Mdepth> float
Range: 0 to 100
Default unit: percent

Example: see `PULSe:MOP:AM:TYPE` on page 338

Manual operation: See "AM" on page 83

PULSe:MOP:AM:TYPE <Type>

Selects the modulation type.

Parameters:
<Type> STD | LSB | USB | SB

Example:
```
PULSe:MOP:TYPE AM
PULSe:MOP:AM:TYPE STD
PULSe:MOP:AM:FREQuency 50000
PULSe:MOP:AM:MDEPth 50
```

Manual operation: See "AM" on page 83

PULSe:MOP:AMSTep:DURation <Duration>
PULSe:MOP:FMSTep:DURation <Duration>

Sets the step time.

Parameters:
<Duration> float
Range: 0 to 3600
Default unit: s

Example: see example "Creating a FM step pulse" on page 332

Manual operation: See "FM Step" on page 86

PULSe:MOP:AMSTep:LEVel <Level>

Sets the step level.

Parameters:
<Level> float
Range: -100 to 0

Example: see example "Creating a FM step pulse" on page 332

Manual operation: See "AM Step" on page 84

PULSe:MOP:FMSTep:FREQuency <Frequency>

Sets the step frequency.

Parameters:
<Frequency> float
Range: -1e+09 to 1e+09

Example: see example "Creating a FM step pulse" on page 332

Manual operation: See "FM Step" on page 86

PULSe:MOP:ASK:INVert <Invert>

Inverts the modulation.

Parameters:
<Invert> ON | OFF | 1 | 0

Example:
```
PULSe:MOP:TYPE ASK
PULSe:MOP:ASK:MDEPth 100
PULSe:MOP:ASK:INVert 1
PULSe:MOP:ASK:SRATe 1e+06
```

Manual operation: See "ASK" on page 83

PULSe:MOP:ASK:MDEPth <Mdepth>

Sets the modulation depth.

Parameters:
<Mdepth> float
Range: 0 to 100
Default unit: percent

Example: see `PULSe:MOP:ASK:INVert` on page 339

Manual operation: See "ASK" on page 83

PULSe:MOP:ASK:SRATe <Srate>

Sets the symbol rate.

Parameters:
<Srate> float
Range: 1 to 1e+09

Example: see `PULSe:MOP:ASK:INVert` on page 339

Manual operation: See "ASK" on page 83

PULSe:MOP:BARKer:CODE <Code>

Selects the code sequence.

Parameters:
<Code> R3 | R4A | R4B | R5 | R7 | R11 | R13

Example:
```
PULSe:MOP:TYPE BARKer
PULSe:MOP:BARKer:CODE R13
PULSe:MOP:BARKer:TTIMe 1
```

Manual operation: See "Barker" on page 89

PULSe:MOP:BARKer:TTIMe <Ttime>

Sets the transistion time.

Parameters:

<Ttime> float

Range: 0 to 50
Default unit: percent

Example: see `PULSe:MOP:BARKer:CODE` on page 339

Manual operation: See "Barker" on page 89

PULSe:MOP:BPSK:PHASe <Phase>

Sets the phase.

Parameters:

<Phase> float

Range: 0.1 to 180
Default unit: degree

Example: see example "Creating a BPSK pulse" on page 333

Manual operation: See "BPSK" on page 91

PULSe:MOP:BPSK:SRATe <Srate>

Sets the symbol rate.

Parameters:

<Srate> float

Range: 1 to 1e+09

Example: see example "Creating a BPSK pulse" on page 333

Manual operation: See "BPSK" on page 91

PULSe:MOP:BPSK:SRATe:AUTO <Auto>

Enables automatic adjusting of the bits in the pulse width.

Parameters:

<Auto> ON | OFF | 1 | 0

Example: see example "Creating a BPSK pulse" on page 333

Manual operation: See "BPSK" on page 91

PULSe:MOP:BPSK:TTIMe <Ttime>

Sets the transition time.

Parameters:
<Ttime> float
Range: 0 to 50
Default unit: percent

Example: see example "Creating a BPSK pulse" on page 333

Manual operation: See "BPSK" on page 91

PULSe:MOP:BPSK:TTYPe <Ttype>

Selects the transition type.

Parameters:
<Ttype> LINear | COSine

Example: see example "Creating a BPSK pulse" on page 333

Manual operation: See "BPSK" on page 91

PULSe:MOP:BPSK:TYPE <Type>

Sets the modulation type.

Parameters:
<Type> STANdard | CONStant

Example: see example "Creating a BPSK pulse" on page 333

Manual operation: See "BPSK" on page 91

PULSe:MOP:CHIRp:DEViation <Deviation>

Sets the modulation deviation.

Parameters:
<Deviation> float
Range: 1 to 1e+09

Example: see example "Creating a linear chirp pulse" on page 332

Manual operation: See "Linear Chirp" on page 87

PULSe:MOP:CHIRp:TYPE <Type>

Selects the modulation type.

Parameters:
<Type> UP | DOWN | SINE | TRIangular

Example: see example "Creating a linear chirp pulse" on page 332

Manual operation: See "Linear Chirp" on page 87

PULSe:MOP:DATA:CODing <Coding>

Selects the data coding scheme.

Parameters:
<Coding> NONE | DIFFerential | GRAY | DGRay

Example: see Using a plugin as a modulaiton source

Manual operation: See "Coding" on page 95

PULSe:MOP:DATA:DSRC <Dsrc>

Selects the data source for the modulation, see `DSRC:CREate`.

Parameters:
<Dsrc> string

Example: see Using a plugin as a modulaiton source

Manual operation: See "Data Source" on page 95

PULSe:MOP:ENABle <Enable>

Defines whether a MOP is applied.

Parameters:
<Enable> ON | OFF | 1 | 0

Example: see example "Creating a linear chirp pulse" on page 332

Manual operation: See "Enable Modulation on Pulse (MOP)" on page 80

PULSe:MOP:EXCLude:ENABle <Enable>

Activates the restriction of the modulation.

Parameters:
<Enable> ON | OFF | 1 | 0

Example: example "Creating an unmodulated pulse" on page 331

Manual operation: See "Restrict modulation to a certain area of the pulse" on page 80

PULSe:MOP:EXCLude:LEVel:STARt <Start>
PULSe:MOP:EXCLude:LEVel:STOP <Stop>

Sets the threshold levels at the beginning and the end of a pulse for the modulation to be excluded.

Parameters:
<Stop> float
Range: 0 to 100

Example: see example "Creating a linear chirp pulse" on page 332

Manual operation: See "Restrict modulation to a certain area of the pulse" on page 80

PULSe:MOP:EXCLude:MODE <Mode>

Selects the parameter that determines the area the MOP is to be restricted.

Parameters:
<Mode> TIME | LEVel | WIDTh

Example: see example "Creating a linear chirp pulse" on page 332

Manual operation: See "Restrict modulation to a certain area of the pulse" on page 80

PULSe:MOP:EXCLude:TIME:STARt <Start>
PULSe:MOP:EXCLude:TIME:STOP <Stop>

Sets a time span to be excluded at the beginning and at the end of the pulse.

Parameters:
<Stop> float
Range: 0 to 5e-05

Example: see example "Creating a linear chirp pulse" on page 332

Manual operation: See "Restrict modulation to a certain area of the pulse" on page 80

PULSe:MOP:FILTer:BT <Bt>

Sets the B x T filter parameter.

Parameters:
<Bt> float
Range: 0.15 to 2.5

Example: see example "Using a plugin as a modulaiton source" on page 326

Manual operation: See "Filter" on page 95

PULSe:MOP:FILTer:BWIDth <Bwidth>

Sets the transition bandwidth of the filter.

Parameters:
<Bwidth> float
Range: 1 to 1e+09

Example: see example "Using a plugin as a modulaiton source" on page 326

Manual operation: See "Filter" on page 95

PULSe:MOP:FILTer:LENGth <Length>

Sets the filter length.

Parameters:
<Length> integer
Range: 4 to 512

Example: see example "Using a plugin as a modulaiton source" on page 326

Manual operation: See "Filter" on page 95

PULSe:MOP:FILTer:ROLLoff <Rolloff>

Sets the roll off factor.

Parameters:
<Rolloff> float
Range: 0.05 to 1

Example: see example "Using a plugin as a modulaiton source" on page 326

Manual operation: See "Filter" on page 95

PULSe:MOP:FILTer:TYPE <Type>

Selects the filter type.

Parameters:
<Type> NONE | RECTangular | COS | RCOS | GAUSs | LPASs

Example: see example "Using a plugin as a modulaiton source" on page 326

Manual operation: See "Filter" on page 95

PULSe:MOP:FM:DEViation <Deviation>

Sets the modulation deviation.

Parameters:
<Deviation> float
Range: 0.1 to 1e+09
Default unit: Hz

Manual operation: See "FM" on page 85

PULSe:MOP:FM:FREQuency <Frequency>

Sets the modulation frequency.

Parameters:
<Frequency> float
Range: 0.002 to 1e+09

Manual operation: See "FM" on page 85

PULSe:MOP:FSK:DEViation <Deviation>

Sets the modulation deviation.

Parameters:
<Deviation> float
Range: 0.001 to 1e+09
Default unit: Hz

Manual operation: See "FSK" on page 85

PULSe:MOP:FSK:SRATe <Srate>

Sets the symbol rate of the modulated signal.

Parameters:
<Srate> float
Range: 1 to 1e+09

Manual operation: See "FSK" on page 85

PULSe:MOP:NLCHirp:EQUation <Equation>

Determines the chirp mathematically.

Parameters:
<Equation> string

Manual operation: See "Equation" on page 88

PULSe:MOP:NOISe:BWIDth <Bwidth>

Sets the bandwidth.

Parameters:
<Bwidth> float
Range: 1 to 1e+09
Default unit: Hz

Manual operation: see chapter 7.2.6.6, "Noise", on page 93

PULSe:MOP:PCHirp:COEFficient <Coefficient>

Sets the coefficient of the chirp polynom.

Parameters:
<Coefficient> float
Range: -1e+22 to 1e+22

Manual operation: See "Polynomial Chirp" on page 88

PULSe:MOP:PCHirp:TERM <Term>

Sets the term of the chirp polynom.

Parameters:
<Term> float
Range: 0 to 32

Manual operation: See "Polynomial Chirp" on page 88

PULSe:MOP:PLISt:VALue <Value>

Sets the phase.

Parameters:
<Value> float
Range: -180 to 180
Default unit: degree

Manual operation: See "Custom Phase" on page 90

PULSe:MOP:POLY:LENGth <Length>

Sets the polyphase length (code order).

Parameters:
<Length> integer
Range: 1 to 100

Manual operation: See "Poly Phase" on page 90

PULSe:MOP:POLY:TYPE <Type>

Selects the modulation type.

Parameters:
<Type> FRANk | P1 | P2 | P3 | P4

Manual operation: See "Poly Phase" on page 90

PULSe:MOP:QPSK:SRATe <Srate>

Sets the symbol rate.

Parameters:
<Srate> float
Range: 1 to 1e+09

Manual operation: See "Symbol Rate" on page 93

PULSe:MOP:QPSK:TYPE <Type>

Selects the modulation type.

Parameters:
<Type> NORMal | OQPSk | DQPSk

Manual operation: See "QPSK Type" on page 93

PULSe:MOP:TYPE <Type>

Select the modulation scheme.

Parameters:
<Type> AM | ASK | AMSTep | FM | FSK | FMSTep | CHIRp | NLCHirp | PCHirp | BARKer | POLYphase | PLISt | BPSK | QPSK | NOISe | PLUGin

Example: see example "Creating a linear chirp pulse" on page 332

Manual operation: see chapter 7.2.6, "Built-In Modulation Types and their Settings", on page 82

PULSe:OVERshoot <Overshoot>

Sets the overshoot level value.

Parameters:
<Overshoot> float
Range: 0 to 50

Example: see example "Creating a linear chirp pulse" on page 332

Manual operation: See "Overshoot" on page 74

PULSe:OVERshoot:DECay <Decay>

Sets the number of peaks.

Parameters:
<Decay> float
Range: 1 to 100

Example: see example "Creating a linear chirp pulse" on page 332

Manual operation: See "Overshoot" on page 74

PULSe:RIPPle <Ripple>

Sets the ripple level.

Parameters:
<Ripple> float
Range: 0 to 50

Example: see example "Creating a linear chirp pulse" on page 332

Manual operation: See "Ripple" on page 75

PULSe:RIPPle:FREQuency <Frequency>

Sets the ripple frequency.

Parameters:
<Frequency> float
Range: 0 to 3e+08
Default unit: Hz

Example: see example "Creating a linear chirp pulse" on page 332

Manual operation: See "Ripple" on page 75

PULSe:TIME:FALL <Fall>
PULSe:TIME:RISE <Rise>

Sets the transition time of the rising and falling edges.

Parameters:
<Rise> float
Range: 0 to 3600

Example: see example "Creating a linear chirp pulse" on page 332

Manual operation: See "Rising/Falling Edge, Width, Rising/Falling Slope" on page 71

PULSe:TIME:REFerence <Reference>

Selects a predefined envelope profile.

Parameters:
<Reference> VOLTage | POWer | FULL

Example: see example "Creating a linear chirp pulse" on page 332

Manual operation: See "Standard Timing Profile" on page 71

PULSe:TIME:WIDTh <Width>

Sets the time the pulse is on top power.

Parameters:

<Width> float

Range: 0 to 3600
Default unit: s

Example: see example "Creating a linear chirp pulse" on page 332

Manual operation: See "Rising/Falling Edge, Width, Rising/Falling Slope" on page 71

PULSe:TYPE:FALL <Fall>
PULSe:TYPE:RISE <Rise>

Sets the slope type for the rising and falling edges.

Parameters:

<Rise> LINear | COSine | RCOSine | SQRT

Example: example "Creating an unmodulated pulse" on page 331

Manual operation: See "Rising/Falling Edge, Width, Rising/Falling Slope" on page 71

PULSe:SETTings <Settings>

Switches between the displayed settings.

Setting parameters:

<Settings> TIMing | LEVel | MOP | MKR | GENeral

Example:

```
PULSe:SETTings MOP
PULSe:PREView:MODE MOP
PULSe:PREView:MOP IQ
```

Usage: Setting only

PULSe:PREView:MODE <Mode>

Switches between the envelope and modulation graphs.

Setting parameters:

<Mode> ENVelope | MOP

Example: see `PULSe:SETTings` on page 349

Usage: Setting only

Manual operation: See "Envelope Graph" on page 96

PULSe:PREView:MOP <Mop>

Sets the displayed modulation characteristics.

Setting parameters:

<Mop> IQ | PHASe | FREQuency

Example: see `PULSe:SETTings` on page 349

Usage: Setting only

Manual operation: See "Time domain display" on page 97

22.15 Repository Commands



Do not change the default folder structure and the repository file names.

Changing file paths and file names may lead to data lost and irreparable faults in the data sources.

Example: Working with repositories

```
SCPI
// create new empty repository
REPository:CREate "Repository for tests"
REPository:CATalog?
//"Repository for tests","C:\Users\Public\Documents\Rohde-Schwarz\Pulse Sequencer\Repositories
//"K300 and K301 Tests","C:\Users\Public\Documents\Rohde-Schwarz\Pulse Sequencer\Repositories"
//"Rep","C:\Users\Public\Documents\Rohde-Schwarz\Pulse Sequencer\Repositories"
REPository:SAVE
REPository:SELect "Repository for tests"
REPository:AUTHor "My Company Name"
REPository:DATE?
REPository:SECurity LEV1
REPository:VERSion "1.1"
REPository:COMPlexity EMITter
REPository:SAVE
REPository:PATH?
// C:\Users\Public\Documents\Rohde-Schwarz\PulseSequencer\Repositories\20140722_113521
REPository:FILename?
// C:\Users\Public\Documents\Rohde-Schwarz\PulseSequencer\Repositories
   \20140722_113521\Config.ps_rep
REPository:ACCess?
// RW,Login=no,Pass=no,Uname=testuser
REPository:REMove "Rep"
```

```
REPManager:PATH:LIST?
// "C:\Users\Public\Documents\Rohde-Schwarz\PulseSequencer\Repositories"
REPManager:PATH:ADD "C:\_ps_files"
REPManager:PATH:LIST?
// "C:\Users\Public\Documents\Rohde-Schwarz\PulseSequencer\Repositories","C:\_ps_files"
REPManager:CATalog?
// "Repository for tests",
   "C:\Users\Public\Documents\Rohde-Schwarz\PulseSequencer\Repositories",
// "K300 and K301 Tests",
   "C:\Users\Public\Documents\Rohde-Schwarz\PulseSequencer\Repositories",
// "My_Tests","C:\_ps_files"
REPManager:LOAD "K300 and K301 Tests"
REPManager:EXPort "K300 and K301 Tests","c:\_ps_files\my.psarch"
REPManager:DISCard "My_Tests"
REPManager:CATalog?
// "Repository for tests",
   "C:\Users\Public\Documents\Rohde-Schwarz\PulseSequencer\Repositories",
// "K300 and K301 Tests",
   "C:\Users\Public\Documents\Rohde-Schwarz\PulseSequencer\Repositories"
```

REPository:ACCess? 351
REPository:AUTHor 352
REPository:COMPlexity 352
REPository:DATE 352
REPository:FILename? 352
REPository:PATH? 353
REPository:SAVE 353
REPository:SECurity 353
REPository:VERSion 353
REPManager:CATalog? 353
REPManager:LOAD 354
REPManager:DISCard 354
REPManager:EXPort 355
REPManager:PATH:ADD 355
REPManager:PATH:DELete 356
REPManager:PATH:LIST? 356

REPository:ACCess?

Queries information on the access rights of the current user.

Return values:

<Access> <permission>,<login>,<pass>,<Uname>

<permission>
Permission of the curret user, e.g. `RW` (read-write)

<login>,<pass>
`Login/Pass=No`: Password not required
`Login/Pass=Yes`: Password required

<Uname>
user name of the current user

Example: see example "Working with repositories" on page 350

Usage: Query only

Manual operation: See "Users Management" on page 43

REPository:AUTHor <Author>

Enters information on the author.

Parameters:
<Author> string

Example: see example "Working with repositories" on page 350

Manual operation: See "General Repository Settings" on page 42

REPository:COMPlexity <Complexity>

Sets the complexity level.

Parameters:
<Complexity> PTRain | EMITter

Example: see example "Working with repositories" on page 350

Manual operation: See "General Repository Settings" on page 42

REPository:DATE <Date>

Queries the creation data.

Parameters:
<Date> string

Example: see example "Working with repositories" on page 350

Manual operation: See "General Repository Settings" on page 42

REPository:FILename?

Queries the file name of the repository archive.

Return values:
<Filename> string
file path, incl. file name and extension

Example: see example "Working with repositories" on page 350

Usage: Query only

Manual operation: See "Storage" on page 44

REPository:PATH?

Queries the directory in that the repository archive is stored.

Return values:
<Path> string

Example: see example "Working with repositories" on page 350

Usage: Query only

Manual operation: See "Storage" on page 44

REPository:SAVE

Stores the repository archive.

To query the storage location, use the command `REPository:PATH?`.

Example: see example "Working with repositories" on page 350

Usage: Event

Manual operation: See "Storage" on page 44

REPository:SECurity <Security>

Sets the security level.

Parameters:
<Security> LEV0 | LEV1 | LEV2 | LEV3 | LEV4

Example: see example "Working with repositories" on page 350

Manual operation: See "General Repository Settings" on page 42

REPository:VERSion <Version>

Sets the repository version.

Parameters:
<Version> string

Example: see example "Working with repositories" on page 350

Manual operation: See "General Repository Settings" on page 42

REPManager:CATalog?

Queries available repository elements in the database.

Return values:

<Catalog> "<RepositryName>","<path>"

`<RepositryName>` is the name of the repository as defined with the command `REPository:CREate`
`<Path>` is the compete file path

Example: see example "Working with repositories" on page 350

Usage: Query only

Manual operation: See "Discovered Repositories on the Mass Storage" on page 47
See "Add Path, Add Install Path, Add Home Path" on page 47

REPManager:LOAD <RepName>[,<Path>[,<Username>,<Passwd>]]

Loads the selected repository to the workspace.

If more than one repositories with the same name exist, loaded is the first repository with a name match.

To query the available repository elements in the database, use the command `REPository:CATalog?`.

Setting parameters:

<RepName> string
Repository name, as configured in the workspace.

<Path> string
Compete file path, as queried with the command `REPManager:PATH:LIST?`.
The `<Path>` must be specified, if `<Username>` and `<Passwd>` are used.

<Username> string
required if the repository is password protected

<Passwd> string
required if the repository is password protected

Example: see example "Working with repositories" on page 350

Usage: Setting only

Manual operation: See "Load" on page 47

REPManager:DISCard <RepName>[,<Path>[,<Username>,<Passwd>]]

Discards the entire repository from the permanent mass storage.

Setting parameters:

<RepName> string
Repository name, as configured in the workspace.
If more than one repositories with the same name exist, the `<Path>` must be specified.

<Path> string
Compete file path, as queried with the command `REPManager:PATH:LIST?`.
The `<Path>` must be specified, if the `<RepName>` is not unique and if `<Username>` and `<Passwd>` are used.

<Username> string
required if the repository is password protected

<Passwd> string
required if the repository is password protected

Example: see example "Working with repositories" on page 350

Usage: Setting only

Manual operation: See "Discard" on page 47

REPManager:EXPort <RepName>[,<Path>],<PSArchiveFile>

Exports the selected repository file to an archive file.

Setting parameters:

<RepName> string
Repository name, as configured in the workspace.

<Path> string
Compete file path, as queried with the command `REPManager:PATH:LIST?`.

<PSArchiveFile> complete file path, incl. file name and extension (`*.psarch`)

Example: see example "Working with repositories" on page 350

Usage: Setting only

Manual operation: See "Export" on page 47

REPManager:PATH:ADD <Add>

Add the selected directory.

Setting parameters:

<Add> string
complete file path

Example: see example "Working with repositories" on page 350

Usage: Setting only

Manual operation: See "Add Path, Add Install Path, Add Home Path" on page 47

REPManager:PATH:DELete <Delete>

Removes the selected file path.

Setting parameters:

<Delete> string
file path

Example: see example "Working with repositories" on page 350

Usage: Setting only

Manual operation: See "Remove Path" on page 48

REPManager:PATH:LIST?

Queries the direcotry in that the repository files are stored.

Return values:

<List> string
compete file path

Example: see example "Working with repositories" on page 350

Usage: Query only

22.16 Scenario Commands

Example: Creating simple pulse train scenario

```
SCPI
SCENario:CREate "SimplePulseTrain"
SCENario:TYPE SEQuence
SCENario:OUTPut:RF:ENABle 1
SCENario:OUTPut:FREQuency 6e+09
SCENario:OUTPut:LEVel -10
SCENario:OUTPut:RESet:ENABle 1
SCENario:OUTPut:RUNMode CONTinious
SEQuence:CATalog?
// "My_S1_PT","My_S2_S1-F1_S1-F2","My_PT_PRI-Stagger","My_PT_F-Hops",
// "My_WV_Seq","My_TestSequence"
SCENario:SEQuence "My_S1_PT"
SCENario:OUTPut:TARGet INSTrument
GENerator:CATalog?
// "My_R&S SMW","My_R&S SMBV","R&S SMW"
SCENario:GENerator "R&S SMW"
SCENario:GENerator:PATH?
// 1
SCENario:OUTPut:REPository:ENABle 1

SCENario:OUTPut:CLOCk:MODE AUTO
SCENario:OUTPut:CLOCk:AUTO:OVERsampling 8
SCENario:OUTPut:CLOCk:AUTO:BORDer 1000000
// SCENario:OUTPut:CLOCk:MODE MAN
// SCENario:OUTPut:CLOCk:USER?
// 1e+07
SCENario:OUTPut:DURation:MODE SEQuence
// SCENario:OUTPut:DURation:TIME 0.10
SCENario:OUTPut:FORMat WV
SCENario:OUTPut:THReshold -100

SCENario:STARt
SCENario:STATe?
// RUN
SYSTem:PROGress?
// 74
SYSTem:PROGress?
// 100
SCENario:CACHe:VOLatile:VALid?
// 1

// SCENario:CACHe:VOLatile:CLEar
// SCENario:STOP
```

Example: Creating sequence collection scenario

```
SCPI
SCENario:CREate "My_PT_Collection"
SCENario:TYPE CSEQuence
SCENario:CSEQuence:ADD
SCENario:CSEQuence:SELect 1
SCENario:CSEQuence:ALIas "PT"
SCENario:CSEQuence "My_S2_S1-F1_S1-F2"
SCENario:CSEQuence:ADD
SCENario:CSEQuence:SELect 2
SCENario:CSEQuence:ALIas "F-Hops"
SCENario:CSEQuence "My_PT_F-Hops"
SCENario:CSEQuence:CURRent 1
...
SCENario:STARt
```

Example: Creating a simple emitter scenario

```
SCPI
SCENario:CREate "SimpleEmitterScenario"
SCENario:TYPE EMITter
EMITter:CATalog?
// "My_EmitterGuidance","TestEmitter"
SCENario:EMITter "My_EmitterGuidance"
SCENario:EMITter:MODE 2
SCENario:EMITter:MODE:BEAM 1
SCENario:EMITter:DIRection:PHI -10
SCENario:EMITter:DIRection:THETa -5
...
SCENario:OUTPut:TARGet FILE
SCENario:OUTPut:PATH "C:\My_ARB_Files"
SCENario:STARt
```

Example: Creating a scenario with multiple emitters and interferes

```
SCPI
SCENario:CREate "LocalizedEmitters"
SCENario:TYPE LOCalized
EMITter:CATalog?
// "My_EmitterGuidance","TestEmitter"
WAVeform:CATalog?
// "My_Waveform","WV_Matlab","CW","AWGN"
ANTenna:CATalog?
// "Isotropic","My_PencilBeam","My_Cosecant","My_PlanarAntenna","Testantenna"
SCAN:CATalog?
// "My_RasterScan","My_Circular","Test Antenna Scan"

SCENario:LOCalized:RECeiver:ANTenna "My_Cosecant"
SCENario:LOCalized:RECeiver:SCAN "My_Circular"
SCENario:LOCalized:RECeiver:GAIN 5
SCENario:LOCalized:RECeiver:ALTitude 1
SCENario:LOCalized:RECeiver:DIRection:BEARing 0
SCENario:LOCalized:RECeiver:DIRection:ELEVation -5

SCENario:LOCalized:ADD
SCENario:LOCalized:SELect?
// 4
SCENario:LOCalized:SELect 5
SCENario:LOCalized:TYPE WAVeform
SCENario:LOCalized:ALIas "I3"
SCENario:LOCalized:WAVeform "My_Waveform"
SCENario:LOCalized:WAVeform:EIRP 50
SCENario:LOCalized:WAVeform:FREQuency 3e+09
SCENario:LOCalized:WAVeform:ANTenna "Testantenna"
SCENario:LOCalized:WAVeform:SCAN "Test Antenna Scan"
SCENario:LOCalized:DIRection:TRACk 1
SCENario:LOCalized:DIRection:ELEVation 0
SCENario:LOCalized:DIRection:BEARing -53.3035
SCENario:LOCalized:LOCation:EAST 2860.34
SCENario:LOCalized:LOCation:NORTh -3837.93
SCENario:LOCalized:DISTance?
// 4786.57

SCENario:LOCalized:ADD
SCENario:LOCalized:SELect 6
SCENario:LOCalized:TYPE EMITter
SCENario:LOCalized:ALIas "E4"
SCENario:LOCalized:EMITter "My_EmitterGuidance"
SCENario:LOCalized:EMITter:MODE 1
SCENario:LOCalized:EMITter:MODE:BEAM 2
SCENario:LOCalized:DIRection:TRACk 1
SCENario:LOCalized:DIRection:ELEVation 0
SCENario:LOCalized:DIRection:BEARing -87.8901
SCENario:LOCalized:LOCation:EAST 253.448
```

```
SCENario:LOCalized:LOCation:NORTh -6879.31
SCENario:LOCalized:DISTance?
// 6883.98

...
SCENario:STARt
```

SCENario:TYPE ... 361
SCENario:STOP ... 361
SCENario:STARt ... 361
SCENario:STATe? ... 361
SCENario:CACHe:VOLatile:CLEar ... 362
SCENario:CACHe:VOLatile:VALid? ... 362
SCENario:GENerator ... 362
SCENario:GENerator:PATH ... 362
SCENario:SEQuence ... 363
SCENario:SEQuence:CLEar ... 363
SCENario:CEMit:ALIas ... 363
SCENario:CSEQuence:ALIas ... 363
SCENario:LOCalized:ALIas ... 363
SCENario:CEMit:CURRent ... 363
SCENario:LOCalized:CURRent ... 363
SCENario:CSEQuence:CURRent ... 363
SCENario:EMITter:DIRection:PHI ... 364
SCENario:CEMit:DIRection:BEARing ... 364
SCENario:LOCalized:RECeiver:DIRection:BEARing ... 364
SCENario:LOCalized:DIRection:BEARing ... 364
SCENario:EMITter:DIRection:THETa ... 364
SCENario:CEMit:DIRection:ELEVation ... 364
SCENario:LOCalized:RECeiver:DIRection:ELEVation ... 364
SCENario:LOCalized:DIRection:ELEVation ... 364
SCENario:EMITter ... 364
SCENario:CEMit:EMITter ... 364
SCENario:LOCalized:EMITter ... 364
SCENario:EMITter:MODE ... 365
SCENario:CEMit:EMITter:MODE ... 365
SCENario:LOCalized:EMITter:MODE ... 365
SCENario:EMITter:MODE:BEAM ... 365
SCENario:LOCalized:EMITter:MODE:BEAM ... 365
SCENario:CEMit:EMITter:MODE:BEAM ... 365
SCENario:LOCalized:DISTance? ... 365
SCENario:LOCalized:DIRection:TRACk ... 366
SCENario:LOCalized:LOCation:EAST ... 366
SCENario:LOCalized:LOCation:NORTh ... 366
SCENario:LOCalized:RECeiver:ALTitude ... 366
SCENario:LOCalized:LOCation:ALTitude ... 366
SCENario:LOCalized:RECeiver:ANTenna ... 366
SCENario:LOCalized:WAVeform:ANTenna ... 366
SCENario:LOCalized:RECeiver:SCAN ... 367
SCENario:LOCalized:WAVeform:SCAN ... 367

SCENario:LOCalized:TYPE 367
SCENario:LOCalized:RECeiver:GAIN 367
SCENario:LOCalized:WAVeform 367
SCENario:LOCalized:WAVeform:EIRP 367
SCENario:LOCalized:WAVeform:FREQuency 368
SCENario:CSEQuence 368
SCENario:CSEQuence:VARiable 368
SCENario:OUTPut:CLOCk:MODE 368
SCENario:OUTPut:CLOCk:AUTO:OVERsampling 369
SCENario:OUTPut:CLOCk:AUTO:BORDer 369
SCENario:OUTPut:CLOCk:USER 369
SCENario:OUTPut:DURation:MODE 369
SCENario:OUTPut:DURation:TIME 370
SCENario:OUTPut:FORMat 370
SCENario:OUTPut:LOOP:COMBine 370
SCENario:OUTPut:FREQuency 370
SCENario:OUTPut:LEVel 370
SCENario:OUTPut:PATH 371
SCENario:OUTPut:REPository:ENABle 371
SCENario:OUTPut:RESet:ENABle 371
SCENario:OUTPut:RF:ENABle 371
SCENario:OUTPut:RUNMode 371
SCENario:OUTPut:TARGet 372
SCENario:OUTPut:THReshold 372
SCENario:VOLatile:VIEW 372

SCENario:TYPE <Type>

Sets the scenario type.

Parameters:

<Type> SEQuence | CSEQuence | EMITter | CEMitter | LOCalized | WAVeform

Example: see example "Creating simple pulse train scenario" on page 357

Manual operation: See "Scenario Type" on page 55

SCENario:STOP
SCENario:STARt

Starts/stops the signal calculation.

Example: see example "Creating simple pulse train scenario" on page 357

Usage: Event

Manual operation: See "Start/Stop, Busy" on page 57

SCENario:STATe?

Queries the current scenario status.

Return values:
<State> IDLE | RUN

Example: see example "Creating simple pulse train scenario" on page 357

Usage: Query only

Manual operation: See "Start/Stop, Busy" on page 57

SCENario:CACHe:VOLatile:CLEar

Deletes the file from the volatile memory.

Example: see example "Creating simple pulse train scenario" on page 357

Usage: Event

Manual operation: See "Volatile, Repository" on page 57

SCENario:CACHe:VOLatile:VALid?

Queries whether the volatile memory contains a valid waveform.

Return values:
<Valid> ON | OFF | 1 | 0

Example: see example "Creating simple pulse train scenario" on page 357

Usage: Query only

Manual operation: See "Volatile, Repository" on page 57

SCENario:GENerator <Generator>

Sets the signal generator.

Parameters:
<Generator> string

use the command `GENerator:CATalog?` to query a list of configured generator profiles.

Example: see example "Creating simple pulse train scenario" on page 357

Manual operation: See "Upload to VSG (Vector Signal Generator)" on page 58

SCENario:GENerator:PATH <Path>

Selects the signal path that will play the generated signal.

Parameters:
<Path> float

Range: 1 to 32

Example: see example "Creating simple pulse train scenario" on page 357

Manual operation: See "Path" on page 57
See "Upload to VSG (Vector Signal Generator)" on page 58

SCENario:SEQuence <Sequence>

Assigns a pulse sequence, see `SEQuence:CATalog?` on page 276.

Parameters:
<Sequence> string

Example: see example "Creating simple pulse train scenario" on page 357

Manual operation: See "Single Sequence and Waveform Scenario Settings" on page 60

SCENario:SEQuence:CLEar

Usage: Event

Manual operation: See "Single Sequence and Waveform Scenario Settings" on page 60

SCENario:CEMit:ALIas <Alias>
SCENario:CSEQuence:ALIas <Alias>
SCENario:LOCalized:ALIas <Alias>

Enters an alias name.

Parameters:
<Alias> string

Example: see example "Creating a scenario with multiple emitters and interferes" on page 359

Manual operation: See "Properties of the selected Emitter/Interferer" on page 207

SCENario:CEMit:CURRent <Current>
SCENario:LOCalized:CURRent <Current>
SCENario:CSEQuence:CURRent <Current>

Sets the sequence/emitter that is currently used by the scenario.

Parameters:
<Current> float

number of the sequence/emmiter in the list with multiple sequences

*RST: 1

Example: see example "Creating sequence collection scenario" on page 358

Manual operation: See "List of sequences" on page 116

SCENario:EMITter:DIRection:PHI <Phi>
SCENario:CEMit:DIRection:BEARing <Bearing>
SCENario:LOCalized:RECeiver:DIRection:BEARing <Bearing>
SCENario:LOCalized:DIRection:BEARing <Bearing>

Sets the bearing.

Parameters:
<Bearing> float
Range: 0 to 360
Default unit: grad

Example: see example "Creating a scenario with multiple emitters and interferes" on page 359

Manual operation: See "Pointing Direction" on page 208

SCENario:EMITter:DIRection:THETa <Theta>
SCENario:CEMit:DIRection:ELEVation <Elevation>
SCENario:LOCalized:RECeiver:DIRection:ELEVation <Elevation>
SCENario:LOCalized:DIRection:ELEVation <Elevation>

Sets the elevation.

Parameters:
<Elevation> float
Range: -90 to 90
Default unit: grad

Example: see example "Creating a scenario with multiple emitters and interferes" on page 359

Manual operation: See "Pointing Direction" on page 208

SCENario:EMITter <Emitter>
SCENario:CEMit:EMITter <Emitter>
SCENario:LOCalized:EMITter <Emitter>

Assigns an existing emitter or an existing waveform, see `WAVeform:CATalog?` on page 276 and `EMITter:CATalog?` on page 276.

Parameters:
<Emitter> string

Example: see example "Creating a scenario with multiple emitters and interferes" on page 359

Manual operation: See "Properties of the selected Emitter/Interferer" on page 207

SCENario:EMITter:MODE <Mode>
SCENario:CEMit:EMITter:MODE <Mode>
SCENario:LOCalized:EMITter:MODE <Mode>

Set the emitter mode.

Parameters:
<Mode> float
Range: 1 to 16

Example: see example "Creating a scenario with multiple emitters and interferes" on page 359

Manual operation: See "Properties of the selected Emitter/Interferer" on page 207

SCENario:EMITter:MODE:BEAM <Beam>
SCENario:LOCalized:EMITter:MODE:BEAM <Beam>

Sets the used beam of the current mode.

Parameters:
<Beam> float
Range: 1 to 32

Example: see example "Creating a scenario with multiple emitters and interferes" on page 359

Manual operation: See "Properties of the selected Emitter/Interferer" on page 207

SCENario:CEMit:EMITter:MODE:BEAM <Beam>

Parameters:
<Beam> float
Range: 1 to 32

Manual operation: See "Properties of the selected Emitter" on page 186

SCENario:LOCalized:DISTance?

Sets the distance to the receiver.

Return values:
<Distance> float
Range: 0 to 1e+09
Default unit: m

Example: see example "Creating a scenario with multiple emitters and interferes" on page 359

Usage: Query only

Manual operation: See "Location" on page 208

SCENario:LOCalized:DIRection:TRACk <Track>

Turns the antenna in the direction of the receiver.

Parameters:
<Track> ON | OFF | 1 | 0

Example: see example "Creating a scenario with multiple emitters and interferes" on page 359

Manual operation: See "Pointing Direction" on page 208

SCENario:LOCalized:LOCation:EAST <East>
SCENario:LOCalized:LOCation:NORTh <North>

Sets the location of the antenna.

Parameters:
<North> float

Range: -1e+06 to 1e+06
Default unit: m

Example: see example "Creating a scenario with multiple emitters and interferes" on page 359

Manual operation: See "Location" on page 208

SCENario:LOCalized:RECeiver:ALTitude <Altitude>
SCENario:LOCalized:LOCation:ALTitude <Altitude>

Sets the altitude of the antenna.

Parameters:
<Altitude> float

Range: -1e+06 to 1e+06

Example: see example "Creating a scenario with multiple emitters and interferes" on page 359

Manual operation: See "Location" on page 208

SCENario:LOCalized:RECeiver:ANTenna <Antenna>
SCENario:LOCalized:WAVeform:ANTenna <Antenna>

Assignes an existing antenna pattern, see `ANTenna:CATalog?` on page 276.

Parameters:
<Antenna> string

Example: see example "Creating a scenario with multiple emitters and interferes" on page 359

Manual operation: See "Properties of the selected Emitter/Interferer" on page 207

SCENario:LOCalized:RECeiver:SCAN <Scan>
SCENario:LOCalized:WAVeform:SCAN <Scan>

Assignes an existing antenna scan, see `SCAN:CATalog?` on page 276.

Parameters:
<Scan> string

Example: see example "Creating a scenario with multiple emitters and interferes" on page 359

Manual operation: See "Properties of the selected Emitter/Interferer" on page 207

SCENario:LOCalized:TYPE <Type>

Defines whether an emitter or an interferer is configured.

Parameters:
<Type> EMITter | WAVeform

Example: see example "Creating a scenario with multiple emitters and interferes" on page 359

Manual operation: See "Properties of the selected Emitter/Interferer" on page 207

SCENario:LOCalized:RECeiver:GAIN <Gain>

Sets the antenna Gain.

Parameters:
<Gain> float
Range: -120 to 120

Example: see example "Creating a scenario with multiple emitters and interferes" on page 359

Manual operation: See "Gain" on page 205

SCENario:LOCalized:WAVeform <Waveform>

Assigns an existing emitter or an existing waveform, see `WAVeform:CATalog?` on page 276 and `EMITter:CATalog?` on page 276.

Parameters:
<Waveform> string

Example: see example "Creating a scenario with multiple emitters and interferes" on page 359

Manual operation: See "Properties of the selected Emitter/Interferer" on page 207

SCENario:LOCalized:WAVeform:EIRP <Eirp>

Sets the EIRP of the interferer.

Parameters:
<Eirp> float
Range: -200 to 200

Example: see example "Creating a scenario with multiple emitters and interferes" on page 359

Manual operation: See "Properties of the selected Emitter/Interferer" on page 207

SCENario:LOCalized:WAVeform:FREQuency <Frequency>

Sets the frequency of the interferer.

Parameters:
<Frequency> float
Range: 1000 to 4.4e+10

Example: see example "Creating a scenario with multiple emitters and interferes" on page 359

Manual operation: See "Properties of the selected Emitter/Interferer" on page 207

SCENario:CSEQuence <Csequence>

Select an exsiting sequence, see `SEQuence:CATalog?` on page 276.

Parameters:
<Csequence> string

Example: see example "Creating sequence collection scenario" on page 358

Manual operation: See "Sequence (Collection) Scenario Settings" on page 61
See "Sequence" on page 117

SCENario:CSEQuence:VARiable <Variable>

Sets the collection variable.

Parameters:
<Variable> string

Example: `SCENario:CSEQuence:VARiable?`

Manual operation: See "Collection Variables" on page 117

SCENario:OUTPut:CLOCk:MODE <Mode>

Sets the clock mode.

Parameters:

<Mode> AUTO | MANual

AUTO
Clock rate is retrived form the generated waveform

MANual
Clock rate is user-defined

Example: see example "Creating simple pulse train scenario" on page 357

Manual operation: See "Clock, Duration" on page 248

SCENario:OUTPut:CLOCk:AUTO:OVERsampling <Oversampling>

Sets the minimum oversampling factor.

Parameters:

<Oversampling> float

Range: 1 to 1000

Example: see example "Creating simple pulse train scenario" on page 357

Manual operation: See "Clock, Duration" on page 248

SCENario:OUTPut:CLOCk:AUTO:BORDer <Border>

Sets the minimum clock rate.

Parameters:

<Border> float

Range: 1000 to 1e+08

Example: see example "Creating simple pulse train scenario" on page 357

Manual operation: See "Clock, Duration" on page 248

SCENario:OUTPut:CLOCk:USER <User>

Sets a user defined clock rate.

Parameters:

<User> float

Range: 1 to 2e+09

Example: see example "Creating simple pulse train scenario" on page 357

Manual operation: See "Clock, Duration" on page 248

SCENario:OUTPut:DURation:MODE <Mode>

Defines the waveform duration.

Parameters:

<Mode> SEQuence | MANual | SCAN

Example: see example "Creating simple pulse train scenario" on page 357

Manual operation: See "Clock, Duration" on page 248

SCENario:OUTPut:DURation:TIME <Time>

Sets the duration of the generated waveform.

Parameters:
<Time> float
Range: 1e-06 to 3600

Example: see example "Creating simple pulse train scenario" on page 357

Manual operation: See "Clock, Duration" on page 248

SCENario:OUTPut:FORMat <Format>

Sets the type of the generated waveform file.

Parameters:
<Format> WV | MSW

Example: see example "Creating simple pulse train scenario" on page 357

Manual operation: See "Output Format" on page 247

SCENario:OUTPut:LOOP:COMBine <Combine>

Combines loop contents into a single segment.

Parameters:
<Combine> ON | OFF | 1 | 0

Manual operation: See "Output Format" on page 247

SCENario:OUTPut:FREQuency <Frequency>

Sets the carrier RF frequency of the generated signal.

Parameters:
<Frequency> float
Range: 1000 to 4.4e+10

Example: see example "Creating simple pulse train scenario" on page 357

Manual operation: See "Frequency" on page 56

SCENario:OUTPut:LEVel <Level>

Sets the reference level used by the calculation of the pulse envelope.

Parameters:

<Level> float
Range: -200 to 30

Example: see example "Creating simple pulse train scenario" on page 357

Manual operation: See "Ref. Level" on page 56

SCENario:OUTPut:PATH <Path>

Sets the directory the generated waveform is stored in.

Parameters:

<Path> string
file path

Example: see example "Creating a simple emitter scenario" on page 358

Manual operation: See "Load ARB File" on page 59

SCENario:OUTPut:REPository:ENABle <Enable>

Enables file storage in the repository.

Parameters:

<Enable> ON | OFF | 1 | 0

Example: see example "Creating simple pulse train scenario" on page 357

Manual operation: See "Volatile, Repository" on page 57

SCENario:OUTPut:RESet:ENABle <Enable>

Restarts the connected instrument on sceanrio start.

Parameters:

<Enable> ON | OFF | 1 | 0

Example: see example "Creating simple pulse train scenario" on page 357

Manual operation: See "Start with Reset" on page 56

SCENario:OUTPut:RF:ENABle <Enable>

Parameters:

<Enable> ON | OFF | 1 | 0

Manual operation: See "Set (RF level and frequency on target instrument)" on page 56

SCENario:OUTPut:RUNMode <Runmode>

Defines the way the generated signal is processed.

Parameters:
<Runmode> CONTinuous | SINGle

Example: see example "Creating simple pulse train scenario" on page 357

Manual operation: See "Playback mode" on page 57

SCENario:OUTPut:TARGet <Target>

Defines whether the software creates an ARB file or transfers the generated waveform to a connected physical generator.

To assign a generator profile, use the command `SCENario:GENerator` on page 362.

To set the name and the directory the ARB file is stored in, use the command `SCENario:OUTPut:PATH`.

Parameters:
<Target> INSTrument | FILE

Firmware/Software: see example "Creating simple pulse train scenario" on page 357

Manual operation: See "Upload to VSG (Vector Signal Generator)" on page 58

SCENario:OUTPut:THReshold <Threshold>

Sets a threshold. Pulses at levels below this threshold are omitted.

Parameters:
<Threshold> float
Range: -100 to 0

Example: see example "Creating simple pulse train scenario" on page 357

Manual operation: See "Features" on page 249

SCENario:VOLatile:VIEW

If a waveform exist in the volatile memory, opens the "Waveform Viewer" and displays this waveform.

Usage: Event

Manual operation: See "Volatile, Repository" on page 57

22.17 Sequence Commands

In the following examples, we assume that a pulse-base sequence "Test Sequence" and the required Pulses have been created.

See also:

- example "Handling items" on page 272

- example "Creating an unmodulated pulse" on page 331

Example: Creating a CW segment

```
SCPI
SEQuence:CREate "Test Sequence"
SEQuence:ITEM:ADD
SEQuence:ITEM COUNt?
// 2
SEQuence:ITEM:SELect 2
SEQuence:ITEM:TYPE FILLer
SEQuence:ITEM:FILLer:SIGNal CW
SEQuence:ITEM:FILLer:MODE DURation
SEQuence:ITEM:FILLer:TIME FIXed
SEQuence:ITEM:FILLer:TIME:FIXed 5e-3
```

Example: Creating a simple sequence with two pulse repeated different number of times

```
SCPI
SEQuence:SELect "Test Sequence"
SEQuence:ITEM:ADD
SEQuence:ITEM:SELect 1; SEQeunce:ITEM:TYPE PULSe
SEQunece:ITEM:PULSe "P1"
SEQuence:ITEM:PRI 2.5 ms
SEQuence:ITEM:PDELay 2 ms
SEQuence:ITEM:REP:TYPE FIXed
SEQuence:ITEM:REP:COUNt:FIXed 2

SEQuence:ITEM:ADD
SEQuence:ITEM:SELect 2; SEQeunce:ITEM:TYPE LOOP
SEQuence:ITEM:LOOP:TYPE VARiable
SEQuence:ITEM:LOOP:COUNt:MINimum 2
SEQuence:ITEM:LOOP:COUNt:MAXimun 10
SEQuence:ITEM:LOOP:COUNt:STEP 2

SEQuence:ITEM:ADD
SEQuence:ITEM:SELect 3; SEQeunce:ITEM:TYPE PULSe
SEQunece:ITEM:PULSe "P1"
SEQuence:ITEM:PRI 5 ms
SEQuence:ITEM:FREQuency:OFFSet 1000000
SEQuence:ITEM:LEVel:OFFSet -10
SEQuence:ITEM:REP:TYPE DURation
SEQuence:ITEM:REP:COUNt:DURation 0.001
SEQuence:ITEM:REP:COUNt:ROUNding UP
SEQuence:ITEM:INDent 1
```

Example: Creating an overlay

```
SCPI
SEQuence:SELect "Test Sequence"
SEQuence:ITEM COUNt?
// 3
SEQuence:ITEM:ADD
SEQuence:ITEM:SELect 4; SEQeunce:ITEM:TYPE PULSe
SEQunece:ITEM:PULSe "P1"
SEQuence:ITEM:PRI 2.5 ms

SEQuence:ITEM:SELect 2
SEQeunce:ITEM:TYPE OVL
SEQuence:ITEM:OVL:WTIMe 0.001

SEQuence:ITEM:SELect 3; SEQuence:ITEM:INDent 1
SEQuence:ITEM:SELect 4; SEQuence:ITEM:INDent 1
```

SEQuence:TYPE 375
SEQuence:ITEM:FILLer:MODE 375
SEQuence:ITEM:FILLer:SIGNal 375
SEQuence:ITEM:FILLer:TIME 375
SEQuence:ITEM:FILLer:TIME:EQUation 375
SEQuence:ITEM:FILLer:TIME:FIXed 376
SEQuence:ITEM:FREQuency:OFFSet 376
SEQuence:ITEM:INDent 376
SEQuence:ITEM:LEVel:OFFSet 376
SEQuence:ITEM:LOOP:COUNt:FIXed 377
SEQuence:ITEM:LOOP:COUNt:MINimum 377
SEQuence:ITEM:LOOP:COUNt:MAXimum 377
SEQuence:ITEM:LOOP:COUNt:STEP 377
SEQuence:ITEM:LOOP:TYPE 377
SEQuence:ITEM:LOOP:VARiable 378
SEQuence:ITEM:OVL:VARiable 378
SEQuence:ITEM:OVL:WTIMe 378
SEQuence:ITEM:PDELay 378
SEQuence:ITEM:PHASe:OFFSet 378
SEQuence:ITEM:PRI 379
SEQuence:ITEM:WAVeform 379
SEQuence:ITEM:PULSe 379
SEQuence:ITEM:REP:COUNt:DURation 379
SEQuence:ITEM:REP:COUNt:FIXed 380
SEQuence:ITEM:REP:COUNt:MINimum 380
SEQuence:ITEM:REP:COUNt:MAXimum 380
SEQuence:ITEM:REP:COUNt:ROUNding 380
SEQuence:ITEM:REP:COUNt:STEP 380
SEQuence:ITEM:REP:TYPE 381
SEQuence:ITEM:REP:VARiable 381
SEQuence:ITEM:TYPE 381

SEQuence:TYPE <Type>

Sets the sequence type.

Parameters:
<Type> PULSe | WAVeform

Example: see example "Handling items" on page 272

Manual operation: See "Sequence Name, Comment and Type, Sequence Description Table, Block Diagram" on page 109

SEQuence:ITEM:FILLer:MODE <Mode>

Sets the way the filler duration is determined.

Parameters:
<Mode> DURation | TSYNc

Example: see example "Creating a CW segment" on page 373

Manual operation: See "Mode" on page 115

SEQuence:ITEM:FILLer:SIGNal <Signal>

Sets the signal type.

Parameters:
<Signal> BLANk | CW | HOLD

Example: see example "Creating a CW segment" on page 373

Manual operation: See "Signal" on page 115

SEQuence:ITEM:FILLer:TIME <Time>

Defiens the way the duration is defined.

Parameters:
<Time> FIXed | EQUation

Example: see example "Creating a CW segment" on page 373

Manual operation: See "Time" on page 115

SEQuence:ITEM:FILLer:TIME:EQUation <Equation>

Sets the filler duration as an equation.

Parameters:
<Equation> string

Example: `SEQuence:ITEM:FILLer:TIME:EQUation?`

Manual operation: See "Time" on page 115

SEQuence:ITEM:FILLer:TIME:FIXed <Fixed>

Sets the duration of the filler.

Parameters:

<Fixed> float

Range: 0 to 3600
Default unit: sec

Example: see example "Creating a CW segment" on page 373

Manual operation: See "Time" on page 115

SEQuence:ITEM:FREQuency:OFFSet <Offset>

Enables a frequency offset.

Parameters:

<Offset> float

Range: -1e+09 to 1e+09
Default unit: Hz

Example: see example "Creating a simple sequence with two pulse repeated different number of times" on page 373

Manual operation: See "Δ Freq." on page 111

SEQuence:ITEM:INDent <Indent>

Indents theselected item rows in order to include it e.g. in a loop.

Parameters:

<Indent> float

Range: 0 to 5

Example: see example "Creating a simple sequence with two pulse repeated different number of times" on page 373

Manual operation: See "Nesting" on page 110

SEQuence:ITEM:LEVel:OFFSet <Offset>

Sets a level offset.

Parameters:

<Offset> float

Range: -100 to 30
Default unit: dB

Example: see example "Creating a simple sequence with two pulse repeated different number of times" on page 373

Manual operation: See "Δ Level" on page 111

SEQuence:ITEM:LOOP:COUNt:FIXed <Fixed>

Sets the repetition number as a numeric value.

Parameters:
<Fixed> float
Range: 1 to 65535

Example: see example "Creating a simple sequence with two pulse repeated different number of times" on page 373

Manual operation: See "Rep. Cont." on page 110

SEQuence:ITEM:LOOP:COUNt:MINimum <Minimum>
SEQuence:ITEM:LOOP:COUNt:MAXimum <Maximum>

Sets the value range of the loop count.

Parameters:
<Maximum> float
Range: 1 to 65535

Example: see example "Creating a simple sequence with two pulse repeated different number of times" on page 373

Manual operation: See "Loop Repetition" on page 113

SEQuence:ITEM:LOOP:COUNt:STEP <Step>

Sets the loop count granularity.

Parameters:
<Step> float
Range: 1 to 65535

Example: see example "Creating a simple sequence with two pulse repeated different number of times" on page 373

Manual operation: See "Loop Repetition" on page 113

SEQuence:ITEM:LOOP:TYPE <Type>

Sets the way the loop repetition is defined.

Parameters:
<Type> FIXed | VARiable

Example: see example "Creating a simple sequence with two pulse repeated different number of times" on page 373

Manual operation: See "Loop Repetition" on page 113

SEQuence:ITEM:LOOP:VARiable <Variable>

Sets a loop variable.

Parameters:
<Variable> string

Example: see example "Creating a simple sequence with two pulse repeated different number of times" on page 373

Manual operation: See "Loop Variables" on page 113

SEQuence:ITEM:OVL:VARiable <Variable>

Sets a variable.

Parameters:
<Variable> string

Manual operation: See "Overlay Variables" on page 114

SEQuence:ITEM:OVL:WTIMe <Wtime>

Sets the duration of the overlay.

Parameters:
<Wtime> float

Range: 0 to 3600
Default unit: sec

Example: see example "Creating an overlay" on page 374

Manual operation: See "Duration" on page 114

SEQuence:ITEM:PDELay <Pdelay>

Enables a start delay.

Parameters:
<Pdelay> float

Range: 0 to 1e+09
Default unit: sec

Example: see example "Creating a simple sequence with two pulse repeated different number of times" on page 373

Manual operation: See "Delay" on page 111

SEQuence:ITEM:PHASe:OFFSet <Offset>

Sets a phase offset.

Parameters:
<Offset> float
Range: -180 to 180

Example: `SEQuence:ITEM:PHASe:OFFSet 30`

Manual operation: See "Phase" on page 111

SEQuence:ITEM:PRI <Pri>

Sets the Pulse Repetition Interval (PRI).

Parameters:
<Pri> float
Range: 0 to 1e+09

Example: see example "Creating a simple sequence with two pulse repeated different number of times" on page 373

Manual operation: See "PRI" on page 111

SEQuence:ITEM:WAVeform <Waveform>
SEQuence:ITEM:PULSe <Pulse>

Assigns a pulse or a waveform to the selected item.

Use the commands `PULSe:CATalog?` and `WAVeform:CATalog?` to querry the available pulses and wavefroms

Parameters:
<Pulse> string
Pulse name

Example: see example "Creating a simple sequence with two pulse repeated different number of times" on page 373

Manual operation: See "Pulse/Waveform" on page 110

SEQuence:ITEM:REP:COUNt:DURation <Duration>

Sets a time duration.

Parameters:
<Duration> float
Range: 0 to 1e+09
Default unit: sec

Example: see example "Creating a simple sequence with two pulse repeated different number of times" on page 373

Manual operation: See "Repetition Number" on page 112

SEQuence:ITEM:REP:COUNt:FIXed <Fixed>

Sets the repetition number as a numeric value.

Parameters:
<Fixed> float
Range: 1 to 65535

Example: see example "Creating a simple sequence with two pulse repeated different number of times" on page 373

Manual operation: See "Rep. Cont." on page 110

SEQuence:ITEM:REP:COUNt:MINimum <Minimum>
SEQuence:ITEM:REP:COUNt:MAXimum <Maximum>

Sets the value range of the repetition count.

Parameters:
<Maximum> float
Range: 1 to 65535

Example: see example "Creating a simple sequence with two pulse repeated different number of times" on page 373

Manual operation: See "Repetition Number" on page 112

SEQuence:ITEM:REP:COUNt:ROUNding <Rounding>

Sets the way the repetition number is rounded.

Parameters:
<Rounding> DN | UP

Example: see example "Creating a simple sequence with two pulse repeated different number of times" on page 373

Manual operation: See "Repetition Number" on page 112

SEQuence:ITEM:REP:COUNt:STEP <Step>

Sets the repetition count granularity.

Parameters:
<Step> float
Range: 1 to 65535

Example: see example "Creating a simple sequence with two pulse repeated different number of times" on page 373

Manual operation: See "Repetition Number" on page 112

SEQuence:ITEM:REP:TYPE <Type>

Sets the way the repetition number is defined.

Parameters:
<Type> FIXed | VARiable | DURation

Example: see example "Creating a simple sequence with two pulse repeated different number of times" on page 373

Manual operation: See "Repetition Number" on page 112

SEQuence:ITEM:REP:VARiable <Variable>

Seta a repetition variable.

Parameters:
<Variable> string

Manual operation: See "Repetition Variables" on page 113

SEQuence:ITEM:TYPE <Type>

Sets the content type of the selected item.

Parameters:
<Type> PULSe | FILLer | LOOP | OVL | SUBSequence | WAVeform

Example: see example "Creating a CW segment" on page 373

Manual operation: See "Type" on page 110

22.18 Status Commands

This system contains the commands for the status reporting system.

Refer to the R&S SMW/SMBV/SGT user manual for detailed information.

Value ranges

- **Return parameters**
  Queries return the current value of the respective register, which permits a check of the device status.
  Range: A decimal value in the range 0 to 32767 (=2^{15}-1)
- **Setting parameters**
  The configuration commands set the respective register thus determining which status changes of the R&S Pulse Sequencer cause the status registers to be changed.
  Range: A decimal value in the range 0 to 32767 (=2^{15}-1)

STATus:OPERation:CONDition?

Queries the content of the CONDition part of the STATus:OPERation register. This part contains information on the action currently being performed in the instrument. The content is not deleted after being read out because it indicates the current status.

Return values:
<Condition> float

Example: `:STATus:OPERation:CONDition?`
queries the Status:Operation:Condition register.

Usage: Query only

22.19 System Commands

The `SYSTem` subsystem contains a series of commands for general functions which do not directly affect signal generation.

SYSTem:ERRor?

Queries the error/event queue for the oldest item and removes it from the queue. The response consists of an error number and a short description of the error.

Positive error numbers are instrument-dependent. Negative error numbers are reserved by the SCPI standard.

Return values:
<Error> string
`Error/event_number, "Error/event_description>[;Device-dependent info]"`
If the queue is empty, the response is `0,"No error"`

Example: see `SYSTem:ERRor:ALL?` on page 382

Usage: Query only

SYSTem:ERRor:ALL?

Queries the error/event queue for all unread items and removes them from the queue. The response is a comma separated list of error number and a short description of the error in FIFO order.

Positive error numbers are instrument-dependent. Negative error numbers are reserved by the SCPI standard.

Return values:

<All> string

List of: `Error/event_number, "Error/event_description>[;Device-dependent info]"`

If the queue is empty, the response is `0,"No error"`

Example: `SYSTem:ERRor:ALL?`

queries all entries in the error queue

`Response: 0, 'no error'`

No errors have occurred since the error queue was last read out

Usage: Query only

SYSTem:ERRor:COUNt?

Queries the number of entries in the error queue. If the error queue is empty, '0' is returned.

Return values:

<Count> string

Example: `SYSTem:ERRor:COUNt?`

queries the number of entries in the error queue

`Response: 1`

One error has occurred since the error queue was last read out

Usage: Query only

SYSTem:PROGress?

Queries the signal generation progress status.

Return values:

<Progress> float

Example: see example "Creating simple pulse train scenario" on page 357

Usage: Query only

Manual operation: See "Start/Stop, Busy" on page 57

22.20 Waveform Commands

Example: Creating waveforms

```
SCPI
WAVeform:CREate "AWGN"
WAVeform:SELect "AWGN"
WAVeform:TYPE NOISe
WAVeform:NOISe:BWIDth 2e+6

WAVeform:CREate "WV"
WAVeform:SELect "WV"
WAVeform:TYPE WAVeform
WAVeform:SIGCont COMM
WAVeform:WAVeform:LOAD "C:\_PS_files\dl_IQ_10MBW.wv"
WAVeform:LEVel:REFerence?
// 14.4695
WAVeform:WAVeform:CLEar
```

WAVeform:TYPE 384
WAVeform:NOISe:BWIDth 384
WAVeform:SIGCont 385
WAVeform:WAVeform:LOAD 385
WAVeform:LEVel:REFerence 385
WAVeform:IQ:CLEar 385
WAVeform:WAVeform:CLEar 385

WAVeform:TYPE <Type>

Sets the type of the waveform.

Parameters:

<Type> CW | NOISe | WAVeform | USER

Example: see example "Creating waveforms" on page 384

Manual operation: See "Type" on page 194

WAVeform:NOISe:BWIDth <Bwidth>

Sets the bandwidth of the generated AWGN waveform.

Parameters:

<Bwidth> float

Range: 0 to 3e+08
Default unit: Hz

Example: see example "Creating waveforms" on page 384

Manual operation: See "Bandwidth" on page 194

WAVeform:SIGCont <Sigcont>

Defines the waveform signal type.

Parameters:
<Sigcont> PULSe | COMM

Example: see example "Creating waveforms" on page 384

Manual operation: See "Signal Content" on page 194

WAVeform:WAVeform:LOAD <Load>

Load the selected file containing a waveform in the Rohde&Schwarz proprietary file format (`*.wv`)

Setting parameters:
<Load> string
complete file path with file name and extenssion

Example: see example "Creating waveforms" on page 384

Usage: Setting only

Manual operation: See "Import" on page 194

WAVeform:LEVel:REFerence <Reference>

Queries the reference level.

Parameters:
<Reference> float
Range: 0 to 100

Example: see example "Creating waveforms" on page 384

Manual operation: See "View, Level" on page 194

WAVeform:IQ:CLEar
WAVeform:WAVeform:CLEar

Removes the imported waveform or file with I/Q data.

Example: see example "Creating waveforms" on page 384

Manual operation: See "Clear" on page 194

22.21 Waveform Viewer Commands

WAVeform:VIEW:XMODe 386
SCENario:VOLatile:VIEW:XMODe 386
WAVeform:VIEW:YMODe 386

SCENario:VOLatile:VIEW:YMODe....386
WAVeform:VIEW:ZOOM:POINt....386
SCENario:VOLatile:VIEW:ZOOM:POINt....386
WAVeform:VIEW:ZOOM:RANGe....387
SCENario:VOLatile:VIEW:ZOOM:RANGe....387

WAVeform:VIEW:XMODe <Xmode>
SCENario:VOLatile:VIEW:XMODe <Xmode>

Sets the units (time or samples) used on the x axis.

Setting parameters:
<Xmode> SAMPles | TIME

Usage: Setting only

Manual operation: See "Units" on page 241

WAVeform:VIEW:YMODe <Ymode>
SCENario:VOLatile:VIEW:YMODe <Ymode>

Sets the view mode.

Setting parameters:
<Ymode> IQ | MAGDb | MAGW | MAGV | PHASe | FREQuency | PAV

Usage: Setting only

Manual operation: See "View mode" on page 241

WAVeform:VIEW:ZOOM:POINt <Point>
SCENario:VOLatile:VIEW:ZOOM:POINt <Point>

Sets center point of the displayed area.

Setting parameters:
<Point> float
always related to time
Default unit: s

Example:
```
SCENario:VOLatile:VIEW:ZOOM:POINt 300 us
SCENario:VOLatile:VIEW:ZOOM:RANGe 100u
// 100 us around the 300 us point, i.e.
// displayed is the time span of 200 us to 400 us

SCENario:VOLatile:VIEW:ZOOM:POINt 0.5 ms
SCENario:VOLatile:VIEW:ZOOM:RANGe 2 m
// displayed is the time span of 0 s to 2 ms
```

Usage: Setting only

Manual operation: See "Zoom in/out" on page 242

WAVeform:VIEW:ZOOM:RANGe <Range>
SCENario:VOLatile:VIEW:ZOOM:RANGe <Range>

Sets the displayed waveform part as a range around the selected center point, set with the command `SCENario:VOLatile:VIEW:ZOOM:POINt`.

Setting parameters:

<Range> float

expressed as a time span (units can be omitted) or as number of samples

Example: see `SCENario:VOLatile:VIEW:ZOOM:POINt`

Usage: Setting only

Manual operation: See "Zoom in/out" on page 242

23 Querying Error Messages and Troubleshooting

The R&S Pulse Sequencer monitors the functions performed and automatically detects errors and irregularities. The software logs status and error messages, warnings, SCPI commands send to the software, or additional information in a log file. Some of the messages are also entered in the error/event queue of the status reporting system.

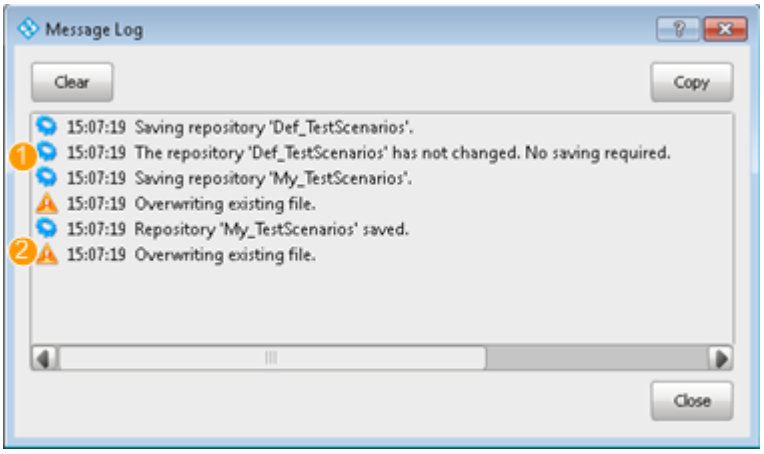
The "Message Log" dialog lists information about all messages in a history list. If the software detects an error, the "Message Log" dialog pops up automatically and displays the error message.

The log information may helps you determine the cause for an error or an unexpected program behavior.

To access the Message Log dialog

► Perform one of the following:

a) In the tool bar, select the "Log" icon.
b) In the menu bar, select "Window > Message Log".



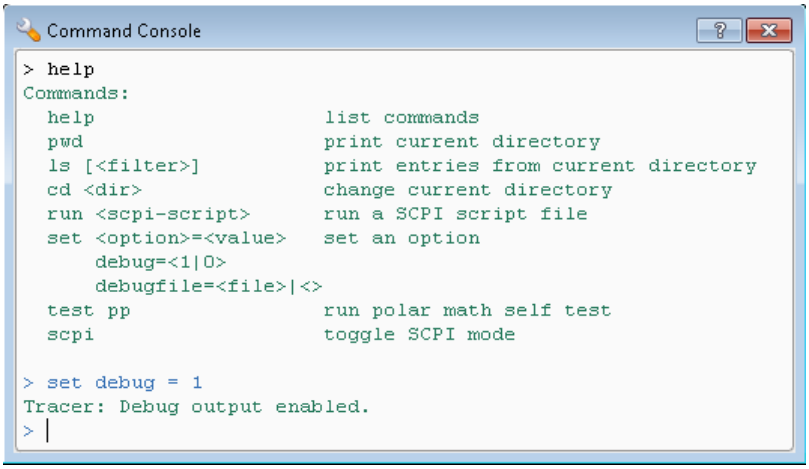
1 = Info message
2 = Warning

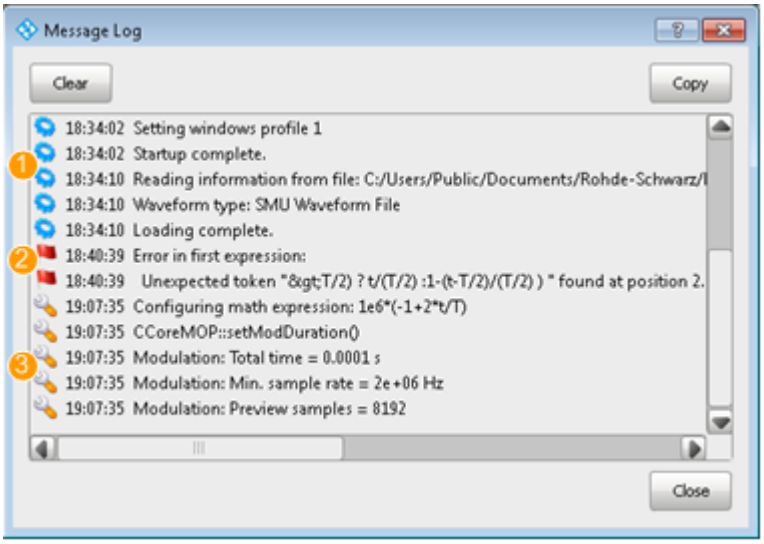
The displayed information is read-only but you can mark and copy it.

To open the Message Log dialog in debug mode

1. In the tool bar, select the "Console Window" icon.
2. In the "Command Console" window, type *help*.

   The dialog lists all available commands.
3. Enter *set debug = 1* to enable the debug messages to be output in the "Message Log" view.




Message Log
Clear
Copy
18:34:02 Setting windows profile 1
18:34:02 Startup complete.
18:34:10 Reading information from file: C:/Users/Public/Documents/Rohde-Schwarz/
18:34:10 Waveform type: SMU Waveform File
18:34:10 Loading complete.
18:40:39 Error in first expression:
18:40:39 Unexpected token ">T/2) ? t/(T/2) :1-(t-T/2)/(T/2)) " found at position 2.
19:07:35 Configuring math expression: 1e6*(-1+2*t/T)
19:07:35 CCoreMOP::setModDuration()
19:07:35 Modulation: Total time = 0.0001 s
19:07:35 Modulation: Min. sample rate = 2e+06 Hz
19:07:35 Modulation: Preview samples = 8192
Close

1 = Info message
2 = Error message
3 = Debug message

To remove write lock that results from a previously crashed session

A red lock symbol on the data base icon in the "Repository Tree" indicates that a session was terminated improperly. A locked repository can be unlocked only from the same PC and by the same user with write permission that had opened the repository before.

1. In the project tree, double click on the repository name.
2. In the "Repository" dialog, select "Storage > Remove Write Lock".

   The repository is unlocked; you can change or store repository settings.
   See also chapter 5.1, "Repository Settings", on page 42.

A Annex

A.1 Supported File Types and File Formats

This section summarizes the file types supported by the application and provides information about the default file locations.

List of supported file types and file extensions

Table 1-1: Supported file types

Extension	Description

`*.ant_pat`
`*.ffe`
`*.tsv`
`*.ffd`
`*.csv`
	Antenna pattern
	See chapter A.1.2, "Antenna Pattern File Formats", on page 393.

`*.dll`	Plugin, e.g. with user modulation, IPM profile
`*.ini`	Initialization file. Contains information on default location for temporary files
`*.wv`	Waveform file in a Rohde&Schwarz proprietary file format. Refer to the user manual of the signal generator for description of this file format
`*.riq`	Rohde&Schwarz proprietary I/Q data file format used by the R&S®PR100 portable receiver.
`*.wav`	Rohde&Schwarz proprietary audio-like file format that contains I/Q samples instead of the usual FM stereo signal. Although the file extension suggest that this is a standard auto file, this file will not be recognized by an stereo player.
`*.iq.tar`	Rohde&Schwarz proprietary I/Q data file format used by some signal and spectrum analyzers, e.g. R&S®FSW. The `*.iq.tar` file contains I/Q data in binary format together with meta information that describes the nature and the source of data, e.g. the sample rate.
`*.mat`	MATLAB file that contain complex 1xN or Mx1 vectors.
`*.txt`	Standard ASCII files with comma separated values. The `*.txt` files are used to describe settings that are based on lists, like for example the custom pulse envelope shapes, the list for IPM profiles, data lists, etc.

A.1.1 File Format of the Reporting Template

The template uses an ASCII text. Text that is not a token or an option (`.OPT`) description is directly transferred to the final report.

Example: Example of a template

```
# R&S Pulse Sequencer, PDW Report
#
Date:       <DATE>
Repository: <REPOSITORY>
Scenario:   <SCENARIO>

TOA      RF      PW      PA      MOP
         GHz     us      dBm
-----------------------------------
.HDR
<TOA> ; <RF> ; <PW> ; <PA> ; <MOP>
# END OF REPORT
.OPT <TOA:12.0f:-9>
.OPT <RF:9.6f:9>
.OPT <PW:7.3f:-6>
.OPT <PA:4.0f:0>
```

Text

<Token>

.HDR
<Col#1> ; <Col#2> ; ...

.OPT <Token:Format:Exponent>

Fig. 1-1: PDW template explanation

General template text and available tokens

The general template text is copied to the final report. All tokens are replaced by values or text provided by the software.

See table 1-2 for an overview of the available tokens.

Table 1-2: List of available tokens

Token name	Description
<USER>	Current user name
<TPATH>	Target file path the generated report file is stored in
<ISODATE>	Date & time in ISO format
<DATE>	Date & time in local format
<REPOSITORY>	Repository name
<RCOMM>	Repository comment
<VERSION>	Repository version
<AUTHOR>	Repository author
<RCDATE>	Repository creation data
<RPATH>	Root path of repository
<SCENARIO>	Scenario name
<SCOMM>	Scenario comment

Adding and formatting data with the .HDR keyword

Each report contains one or more data lines. The position of these data lines in the report is indicated with the keyword `.HDR`. Usually, the reported data is formatted in table form, where the column names are defined with the information right after the `.HDR` line.

Each report template must have a `.HDR` line and subsequent second line defining the individual columns information and following the syntax:

```
.HDR
<Col#1>;<Col#2>;...
```

See table 1-3 for an overview of the available tokens.

Table 1-3: List of available .HDR tokens

Token name	Description
Values related to the generated pulse data	
<TOA>	Time of arrival
<PRI>	PRI of current pulse (0% - 0%)
<PRF>	PRF of current pulse (1/PRI or 0)
<PW>	Pulse width (0% - 0%)
<RT>	Rise time
<FT>	Fall time
<RF>	Center frequency of pulse (absolute)
<BW>	RF bandwidth (FMOP) incl. BB filter
<PA>	Pulse amplitude (at pulse start)
<MF>	MOP flag, where 1 indicates that MOP is active
<MOP>	MOP type, AMOP, PMOD, FMOP
<DF>	Frequency offset from carrier
Global variables from IPM profiles or loops, etc.	
<$<VarName>>	Variable from the internal variable pool
Automatically generated values related to the sequencing	
<@ITEM>	Sequence Line Item, 1... N
<@REP>	Repetition, 1...M
<@COLL>	Collection Line Item, 1...L
<@RFC>	Center Frequency of Emitter / Sequence

Defining option with the .OPT keyword

This keyword defines one or multiple options, that further control the output of the PDW report generator. Each option line must begin with `.OPT` and follow the syntax:

```
.OPT <Token:Format:Exponent>
```

The `.OPT` keyword formats the numeric data output, see table 1-4.

Table 1-4: Option keyword syntax description

Parameter	Description
`Token`	Name of the token
`Format`	The format of the numeric data is compared to a regular expression: [0-9]*.?[0-9]*[-+diufFeExX] To prevent e.g. application crash, data that does not match this expression is ignored.
`Exponent`	Sets an exponent to convert the numeric value; e.g. -9 converts the reported data into nano.

See figure 1-1.

A.1.2 Antenna Pattern File Formats

R&S Pulse Sequencer supports:

- "FEKO far field file format *.ffe" on page 393
- "Antenna Magus file format *.tsv" on page 393
- "ANSYS HFSS file format *.ffd" on page 394
- "Antenna patterns in *.csv file format" on page 394
- "Rohde&Schwarz proprietary antenna pattern *.ant_pat file format" on page 395

FEKO far field file format *.ffe

FEKO is an electromagnetic simulation software tool distributed by the EM Software & Systems-S.A. (Pty) Ltd.

The R&S Pulse Sequencer supports the `*.ffe` files with version V1, V2 and V3.

For information on the file format, see the official product page http://www.feko.info/.

Antenna Magus file format *.tsv

Antenna Magus is an antenna design software tool.

The `*.tsv` files are text files that describe the antenna pattern in a tabular structure. These file contain a header section and data part with tab-separated values.

For information, see http://www.antennamagus.com/.

Example: Antenna Magus *.tsv file (extract)

```
# Far-field exported from Antenna Magus 5.3.0.1686
# On Friday, March 06, 2015 at 1:18:13 PM
# Frequency: 3.0E+9 Hz
# Power: 0.171146799151093 W
# Theta Samples: 181
# Phi Samples: 361
#Theta    Phi    Re(E_Theta)    Im(E_Theta)    Re(E_Phi)    Im(E_Phi)
0.000000000000E+000    0.000000000000E+000    4.303687774512E-016    0.000000000000E+000    -7.148424634567E+000    -1.436268526906E+000
1.000000000000E+000    0.000000000000E+000    4.425352450194E-016    0.000000000000E+000    -7.363209895869E+000    -1.542584165100E+000
...
```

ANSYS HFSS file format *.ffd

ANSYS HFSS software is the industry standard for simulating 3-D full-wave electromagnetic fields.

The R&S Pulse Sequencer supports the `*.ffd` multi-frequency far field pattern files but imports only the first frequency.

For information, see http://www.ansys.com/.

Example: ANSYS HFSS *.ffd file (extract)

The values in the columns are the complex values of the electromagnetic field in Theta and Phi direction.

```
0 180 19
0 360 37
Frequencies 1
Frequency 8.680000000000000e+008
-3.029304169961667e+000 -3.592562203897908e+000 6.108290737961166e-001 1.376043194687679e+000
-2.877212877279787e+000 -3.299035718428788e+000 1.127582356352358e+000 1.978979886470077e+000
```

Antenna patterns in *.csv file format

Several of the popular antenna design software tools export antenna patterns also as `*.csv` files.

The `*.csv` files are text files that describe the antenna pattern in a tabular structure. These files contain a header section and data part with up to six columns. The values can be separated by commas, semicolons or withe spaces.

When a file is selected, the "Import Wizard" tries to automatically determine the used column separator, decimal delimiter and to data content of the columns. You can change the values afterwards.

Example: Antenna Magus file in *.csv format (extract)

```
FRQ=3.0E+9
PIN=0.171146799151093
THETA_SYM_0=0
THETA_SYM_90=0
PHI_SYM_0=0
PHI_SYM_90=0
INTERP_DBI=0
Theta(,deg) Phi(,deg) E_Theta(Re,) E_Theta(Im,) E_Phi(Re,) E_Phi(Im,)

# Far-field exported from Antenna Magus 5.3.0.1686
# On Friday, March 06, 2015 at 1:17:55 PM
# Theta Samples: 181
# Phi Samples: 361
0.000000000000E+000,0.000000000000E+000,4.303687774512E-016,0.000000000000E+000,-7.148424634567E+000,-1.436268526906E+000
1.000000000000E+000,0.000000000000E+000,4.425352450194E-016,0.000000000000E+000,-7.363209895869E+000,-1.542584165100E+000
...
```

Example: ANSYS HFSS files in *.csv format (extract)

The following are two examples of the ANSYS HFSS file in `*.csv` file format. The first file describes the electromagnetic fields, whereas the second one the gain in dBi.

```
Phi[deg],Theta[deg],re(rEPhi)[mV],im(rEPhi)[mV]
0,0,610.829,1376.04
10,0,1127.58,1978.98
...

Phi[deg],Theta[deg],dB(GainTotal)
0,0,-2.79687
10,0,-2.79687
...
```

Rohde&Schwarz proprietary antenna pattern *.ant_pat file format

The table 1-5 describes the used tags and parameters.

*Table 1-5: Format of *.ant_pat file*

Element	Description
<antenna_pattern>	Root element of the antenna pattern file
<az_res>	Resolution of the columns in the `<data>` section value in degrees integer divider of 360
<elev_res>	Resolution of the rows in the `<data>` section value in degrees integer divider of 180
<data>	The file has to contain up to: • [1 + 360/`<az_res>`] columns • [1 + 180/`<elev_res>`] rows If the column resolution exceeds the resolution specified with the `<az_res>` element, an interpolation is applied. You may also define a subset of values, for example to define a beam antenna pattern. Missing values are internally set to zero, see example "Beam antenna pattern" on page 396.

The following are two examples of the file format: the description of a beam-like antenna pattern and the description of an antenna pattern with back lobes and medium resolution.

Example: Beam antenna pattern

This example shows that partially defined antenna patterns are also allowed. In

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<antenna_pattern>
 <az_res> 5.00000000e+00 </az_res>
 <elev_res> 5.00000000e+00 </elev_res>
 <data>
        -30,-20,-10,0,  10, 20, 30
    -30,100,100,100,100,100,100,100,
    -20,100,100,100,100,100,100,100,
    -10,100,100,10, 100,10, 100,100,
      0,100,100,100,0,  100,100,100,
     10,100,100,10, 100,10,1 00,100,
     20,100,100,100,100,100,100,100,
     30,100,100,100,100,100,100,100,
 </data>
</antenna_pattern>
```

the example, defined are only the subset of values that describe the beam pattern; all other values are assumed to be zero, see figure 1-2.

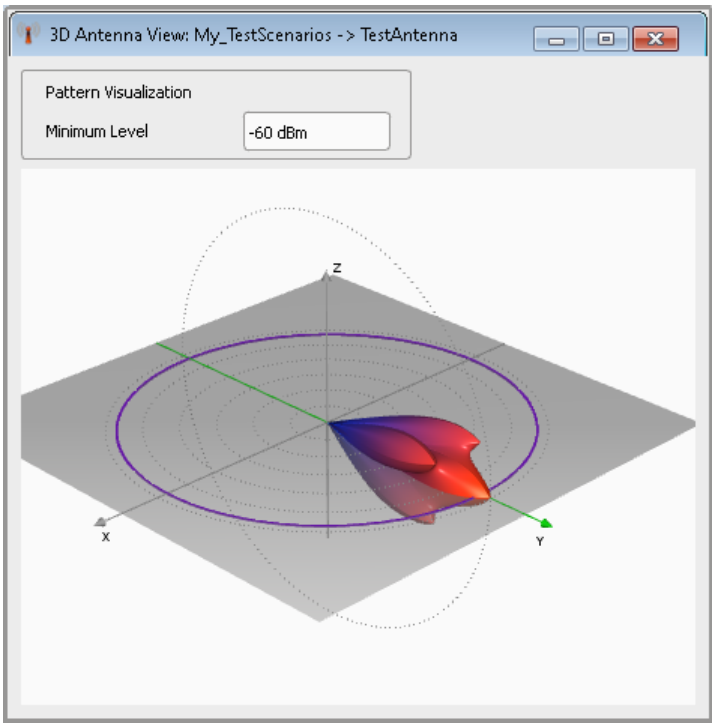


Fig. 1-2: 3D view of the imported partially defined antenna pattern

Example: Antenna pattern with back lobes

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<antenna_pattern>
<az_res> 5.00000000e+00 </az_res>
<elev_res> 5.00000000e+00 </elev_res>
<data>
 -180,-160,-140,-120,-100,-80,-60,-40,-20,0,20, 40, 60, 80, 100,120,140,160,180
 -90,100,100,100,100,100,100,100,100,100,30,100,100,100,100,100,100,100,100,100
 -80,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100
 -70,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100
 -60,100,100,100,100,100,100,100,100,100,20, 100,100,100,100,100,100,100,100,100
 -50,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100
 -40,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100
 -30,100,100,100,100,100,100,100,100,100,10, 100,100,100,100,100,100,100,100,100
 -20,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100
 -10,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100
 0, 30, 100,100,20 ,100,100,10, 100,100,0,  100,100,10, 100,100,20, 100,100,30
 10,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100
 20,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100
 30,100,100,100,100,100,100,100,100,100,10, 100,100,100,100,100,100,100,100,100
 40,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100
 50,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100
 60,100,100,100,100,100,100,100,100,100,20, 100,100,100,100,100,100,100,100,100
 70,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100
 80,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100
 90,100,100,100,100,100,100,100,100,100,30, 100,100,100,100,100,100,100,100,100
</data>
</antenna_pattern>
```

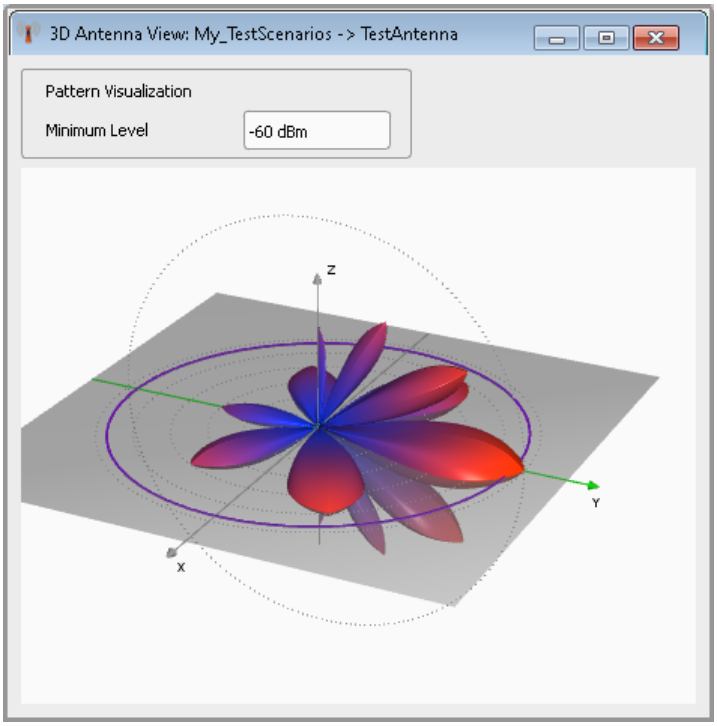


Fig. 1-3: 3D view of the imported antenna pattern with back lobes

A.2 Formula Syntax

You can use mathematical expression to define custom modulation types or envelope shapes.

The R&S Pulse Sequencer uses the fast math parser library muParser, that is an extensible high performance math expression parser library written in C++.

This section list some of the default features supported by the parser. For detailed information, see the product home page http://muparser.beltoforion.de/index.html.

Basic syntax elements

Table 1-6: Mathematical operations

Element	Description
+	Addition
-	Subtraction
*	Multiplication
/	Division
^	By the power of

Table 1-7: Built-in functions (Extract)

Element	Description
sin	sine function
cos	cosine function
exp	raised to the power of x
etc.	

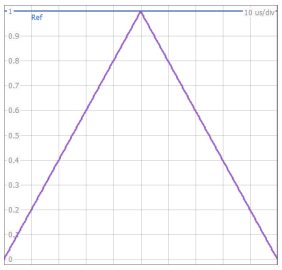
Table 1-8: Other operators

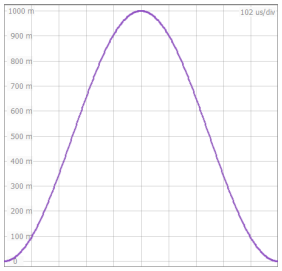
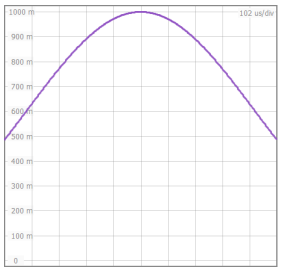
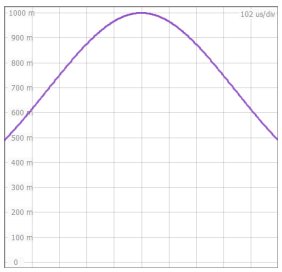
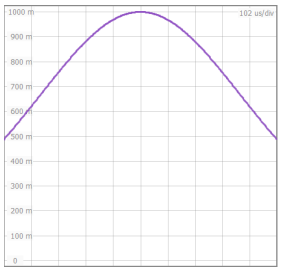
Element	Description	Syntax
?:	if then else operator	`<if_condition>?<then_expression>:<else_expression>`

Table 1-9: Constant and variables

Element		Description
Constants:	PI	π
	T_r	Pulse rise time
	T_w	Pulse width time
	T_f	Pulse fall time
	T $T = T_r+T_w+T_f$	Total pulse time
Variables:	ip<n>	inter-pulse modulation value <n>
	t t = 0 .. T	time

Equations and parameters used to define custom pulse envelopes

Envelope name	Equation	Envelope shape
"Rectangular Pulse"	1	
"Triangular Pulse"	(t<T/2) ? t/(T/2) :1-(t-T/2)/(T/2)	

Envelope name	Equation	Envelope shape
"Raised Cosine Pulse"	0.5 * (1 - cos(2 * PI * t/T))	
"Gaussian Pulse"	exp(-4 * ln(2) * ((t-Tr-Tw/2) / Tw)^2)	
"Lorentzian Pulse"	1 / ((1 + 4 * (sqrt(2)-1) * ((t-Tr-Tw/2) / Tw)^2))^2	
"Sech Pulse"	1 / (cosh((2 * (t-Tr-Tw/2) * ln(1+sqrt(2))) / Tw))^2	

where Amplitude value range is 0 ..1 V.

A.3 Plugin Programming API

The functionality of the R&S Pulse Sequencer can be extended by Plugins. A plugin is a Microsoft Windows DLL. This DLL can be loaded into a repository, and then remains there.

You can load Plugins to:

- specify your custom modulation scheme (MOP)
- define a custom inter-pulse modulation profile (IPM)

- create a report file

The following is a list of all functions that the plugin have to provide.

- Common Functions....401
- MOP Functions....404
- IPM Functions....409

A.3.1 Common Functions

void PS_PLUGIN_EXPORTS getType....401
void PS_PLUGIN_EXPORTS getVersion....401
void PS_PLUGIN_EXPORTS getComment....401
void PS_PLUGIN_EXPORTS getAuthor....402
void PS_PLUGIN_EXPORTS getName....402
void PS_PLUGIN_EXPORTS getError....402
int PS_PLUGIN_EXPORTS getNextMsg....402
int PS_PLUGIN_EXPORTS initPlugin....403
void PS_PLUGIN_EXPORTS shutdownPlugin....403
int PS_PLUGIN_EXPORTS getVariable....403
int PS_PLUGIN_EXPORTS setVariable....404

---

void PS_PLUGIN_EXPORTS getType (char szModType[1024])

(mandatory)

Queries the plugin type.

Return values:

szModType — string

modulation / mop
Plugin for MOP

ipm
Plugin for IPM

report
Plugin for reports created during the waveform generation

---

void PS_PLUGIN_EXPORTS getVersion (char szModVer[1024])

(mandatory)

Queries the version string of the plugin.

Return values:

szModVer — <Major>.<Minor>.<Revision>

`<Major>`, `<Minor>` and `<Revision>` are numerical values

---

void PS_PLUGIN_EXPORTS getComment (char szModComment[4096])

(mandatory)

Queries the comment string of the plugin.

Return values:

szModComment string
Allowed are a single line, a multiple line or an empty string

---

void PS_PLUGIN_EXPORTS getAuthor (char szModAuthor[1024])

(mandatory)

Queries the author string of the plugin.

Return values:

szModAuthor string
Empty string is allowed

---

void PS_PLUGIN_EXPORTS getName (char szModName[1024])

(mandatory)

Queries the plugin name.

Return values:

szModName string
Must not be an empty string. Name must be unique.

---

void PS_PLUGIN_EXPORTS getError (char szModError[1024])

(mandatory)

Queries the plugin errors. This function is started automatically whenever another function returns false and may return additional error information.

Clear the internal error text after receiving the error information.

Return values:

szModError string
Explaining error text.

---

int PS_PLUGIN_EXPORTS getNextMsg char szInfoMsg[4096]

(optional)

Sends messages to the application. The messages are passed to the Message Log of the application, merely for user information.

The function is called as long as it does not return false (0).

Setting parameters:

szInfoMsg "<Prefix>:<Text>"

The massage may contain one of the following prefix:
`D` = Debug
`I` = Information
`W` = Warning
`E` = Error
If no prefix is present, information is assumed.

Usage: Setting only

---

int PS_PLUGIN_EXPORTS initPlugin (void)

(optional)

Initializes the plug-in. It is called once after the plugin is loaded into memory and may set up internal variables.

Return values:

initPlugin boolean

true
The initialization completed successfully

false
Error during the initialization. The plugin is removed from memory

---

void PS_PLUGIN_EXPORTS shutdownPlugin (void)

(optional)

Uninstalls the plugin.

It is called when the main application terminates and may be used to clean up previously allocated memory.

---

int PS_PLUGIN_EXPORTS getVariable (int iIndex, char szConfig[4096])

(optional)

Queries the variables to be registered.

Plugins can register variable sets. Variables are initialized with a default value but can be changed subsequently. If changed, variables are sent back to the plugin before calculation. That is, a plugin can be used with different configurations.

The function is called automatically multiple times; the `iIndex` is increased each time.

Query parameters:

szConfig — The string of variables to be registered is composed as follows:

DBL,<name>,<unit>,<default>,<min>,<max>,<dec. digits>
Real Number

INT,<name>,<unit>,<default>,<min>,<max>
Integer

BOOL,<name>,<default>
Boolean

STR,<name>,<default>[,<validator-regexp>]
String

SEL,<name>,<opt>|<opt>|...,<default index>
Selection
Value Range of `<default index>`: 0 to (number of <opt>-1)

Return values:

iIndex

0
Initial index
When returned latter, `0` indicates that there are no more variables to be registered

> 0
Indicates a valid variable configuration string that has to be registered

Range: 0 to 128

Example:

```
SEL,Type,16-QAM|32-QAM|64-QAM|256-QAM,2
```

This field is named "Type" and lists the options:
16-QAM, 32-QAM, 64-QAM and 128-QAM
The default value is 64-QAM

---

int PS_PLUGIN_EXPORTS setVariable (int iIndex, const char szValue[4096])

(optional)

Sets the values of the variables before a pulse calculation is started.

Parameters:

iIndex — Is the `iIndex` value form the getVariable() function.

szValue — string

Variable value; the string is internally converted.

A.3.2 MOP Functions

int PS_PLUGIN_EXPORTS isDataNeeded 405
double PS_PLUGIN_EXPORTS getModulationClockHint 405
double PS_PLUGIN_EXPORTS getModulationSymbolRate 405
void PS_PLUGIN_EXPORTS setModulationParameter 405
int PS_PLUGIN_EXPORTS getBitsNeeded 406
void PS_PLUGIN_EXPORTS setBits 406

int PS_PLUGIN_EXPORTS modulationCore....406
void PS_PLUGIN_EXPORTS generateDirac....407
int PS_PLUGIN_EXPORTS setGlobalVariable....408
int PS_PLUGIN_EXPORTS getNextReportVariable....408

int PS_PLUGIN_EXPORTS isDataNeeded (void)

(optional)

Queries whether this plugin requires data from a data source or not.

Return values:

isDataNeeded **1**
the plugin requires a data source
*RST: 0

double PS_PLUGIN_EXPORTS getModulationClockHint (void)

(optional)

Queries the minimum clock rate required by the plugin to calculate the I and Q data.

The function is automatically called after the parametrization.

Return values:

getModClock integer
The default value is used automatically, if this function is not implemented or the supplied value is outside the limits
Range: 0 to 10 GHz
*RST: 100 MHz

double PS_PLUGIN_EXPORTS getModulationSymbolRate (void)

(optional)

Queries the symbol rate.

The symbol rate is necessary if data is requested from the data source, e.g. if a digital modulation is implemented; or if a baseband filter is used.

Return values:

getModSymRate integer
The default value is used automatically, if this function is not implemented or the supplied value is outside the limits
Range: 0 to 10 GHz
*RST: 0 Hz

void PS_PLUGIN_EXPORTS setModulationParameter (const char *szType, const char szValue[1024])

(mandatory)

Sets various parameters before a MOP calculation starts.

Parameters:

szType	string identifies the MOP parameter **"mtime"** MOP duration **„srat“** Sample rate, used to calculate the MOP **„done“** Dummy Indicates the last MOP parameter
szValue	string MOP parameter value; the string has to be converted.

---

int PS_PLUGIN_EXPORTS getBitsNeeded (double dTime, long *pcStart, int *pcCount)

(optional)

Requests a number of bits from the data source.

The function is called automatically by the main application before a sample is calculated.

Setting parameters:

dTime	Time stamp of the following sample

Return values:

pcStart	Index of the first bit from the data source
pcCount	Number of required bits

---

void PS_PLUGIN_EXPORTS setBits (long lStart, int iCount, const char *cBuffer)

(optional)

Answers the function getBitsNeeded() and transmits the requested number bits to the plugin.

Return values:

lStart	Start index of the first bit; should match the `pcStart` value
iCount	Number of transmitted bits
cBuffer	Indicates the buffer the bits are transferred to

---

int PS_PLUGIN_EXPORTS modulationCore (double dTime, double *pdAmplitude, double *pdPhase, short *puiMkr)

(mandatory)

Transforms samples into I/Q data.

Parameters:

puiMkr — 32-bit unsigned integer bit field marker flags
Bit 0= Marker 1
Bit 1= Marker 2
Bit 3= Marker 3
Bit 4= Marker 4

Setting parameters:

dTime — Time stamp of the sample to be calculated

pdAmplitude — Predefined amplitude of the last samples

pdPhase — Predefined phase of the last samples
Default unit: rad

Return values:

pdAmplitude — Amplitude, calculated as sqrt(i^2+q^2)
Range: 0.1 ... 1.0

pdPhase — Phase
Default unit: rad

---

void PS_PLUGIN_EXPORTS generateDirac (int *pcGenI, int *pcGenQ)

(optional)

Generates a dirac impulse in the main application for the last sample.

If the function is implemented it gets called after every call to modulationCore ().

The functionality provided by this function is used if the sample returned by `modulationCore ()` is to be folded with a baseband filter. This, for example, is required if the `modulationCore ()` provides samples used with a digital I/Q modulation such as QPSK or QAM. Typically the output signal is created with oversampling factors of two or larger, thus multiple I/Q samples are generated for each symbol. The folding operation, however, only requires one single sample per symbol as a dirac impulse. This function indicates which sample is used as the dirac impulse.

The I and Q channel can be controlled separately. This is required if the constellation points rotate around the origin. In this case the I and Q waveform are time delayed by a fraction of the symbol duration.

If the function is not implemented all generated samples are assumed to be dirac pulses. This is equal to a sample and hold operation.

Return values:

*pcGenI, *pcGenQ — The return value for `*pcGenI` and `*pcGenQ` is true (1) if the last sample shall be used as the dirac pulse.
In all other cases the main application replaces the sample value by zero.

int PS_PLUGIN_EXPORTS setGlobalVariable (const char szType[256], const char szValue[256])

(optional)

Returns a global variable value to the plug-in.

Global variables can be defined in the IPM profiles or in the sequences.

Parameters:

szType — string

Variable names always begin with a $ (dollar sign).
The following variables are predefined:

$RFC
RF center frequency

$LEV
RF level, in dBm

$REP
Repetition count

$ITEM
Item number in a sequence

szValue — string

Value of the variable

Usage: Setting only

---

int PS_PLUGIN_EXPORTS getNextReportVariable (int iIndex, const char szName[256], const char szValue[256])

(optional)

Queries the report variables after a successful calculation and sends them to the report generator.

Parameters:

iIndex

0
Initial index

> 0
Function is called as long as false (0) is returned.

Return values:

szName — string

Variable names, without the $ (dollar sign).

szValue — string

Variable value

A.3.3 IPM Functions

void PS_PLUGIN_EXPORTS setRndFnPtr....409
void PS_PLUGIN_EXPORTS restart....409
int PS_PLUGIN_EXPORTS calculateNextIpmValue....409

void PS_PLUGIN_EXPORTS setRndFnPtr (void *pRndFn, void *pThis)

(optional)

Sets a pointer to a random function `pRndFn` and the `pThis` pointer, which must be specified when using the Random function.

Plugins have to use the random generator provided by the application. This generator produces a random sequence with a defined start seed. If necessary, the application can reproduce a particular calculation with exactly the same values.

The prototype of the callback function is defined as follows:

```
typedef int (*RndFnCallback)( void*, int );
```

To call it:

```
int iRnd = pRndFn( pThis, iMaxVal );
```

The random generator returns a 32-bit random integer value in the range `0 ... iMaxVal`.

void PS_PLUGIN_EXPORTS restart ( )

(optional)

Presets the IPM pattern.

For example, if an IPM list pattern is used, this function sets the start index to first list entry.

int PS_PLUGIN_EXPORTS calculateNextIpmValue (double dTime, unsigned __int64 iCount, double *pdValue)

(mandatory)

Requests the next IPM value.

Setting parameters:

dTime	Time stamp of the next IPM value

Return values:

iCount	64-bit value I/Q sample of the next IPM value Range: 0 to (2^{64}-1)
pdValue	Value of the next IPM

A.4 Common Coding Algorithms

In the notation used below a_n denotes the nth input symbol and b_n denotes the correspondingly coded output symbol. Individual bits in the symbols from the LSB (least significant bit) to the MSB (most significant bit) are denoted by a_{0n}, a_{1n}, etc. The same applies to the output symbols.

Common coding types are listed in table 1-10.

Table 1-10: Common coding algorithms

Coding	Coding algorithm	Applicable for K bit/symbol
"None"	$b_n = a_n$	k = 1...8
"Differential"	$b_n = (a_n + b_{-1})$ modulo 2^k	k = 1...7
"Differential + Gray"	Gray coding with additional differential coding	k = 1...7

The following example illustrates how one of common coding schemes in combination with a modulation method influence the signal.

Example: Differential coding for QPSK modulation with K = 2 bit/symbol

Decimal display; the value range for modulation symbols is $a_n = \{0, 1, 2, 3\}$

According to table 1-10 and for $k = 2$, the recursive coding is defined as $b_n = (a_n + b_{-1})$ *modulo 4*

Depending on the state of a preceding modulation symbol b_{-1} the coded modulation symbol b_n is obtained for example from modulation symbol $\mathbf{a_n = 2}$ as follows:

$\mathbf{b_{-1}}$	0	1	2	3
$\mathbf{b_n}$	2	3	0	1

By means of differential coding, the assignment of modulation symbols a_n (binary indication: MSB, LSB) to the phase differences shown in the following table is generated:

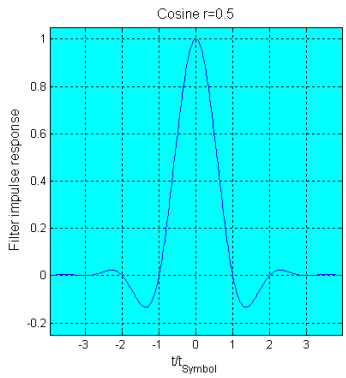
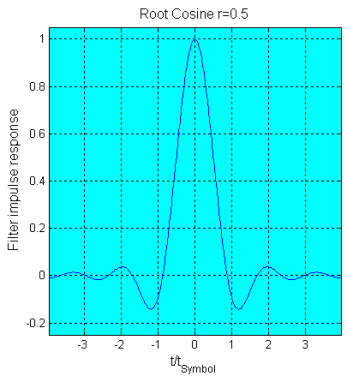
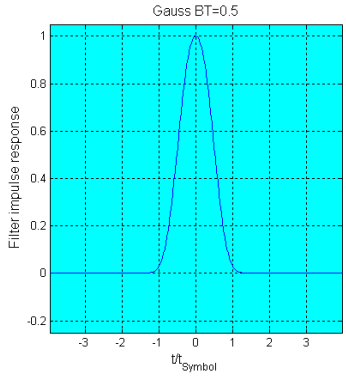
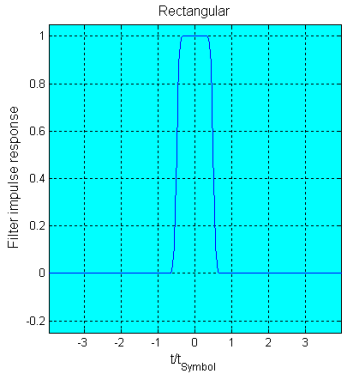
Table 1-11: Phase difference for QPSK

Modulation symbol a_n	00	01	10	11
Phase difference	0°	90°	180°	270°

A.5 Supported Filter and Impact of the Filter Parameters

The table 1-12 shows the filters that are available, together with their associated parameters. The filter characteristic is displayed in graphical form.

Table 1-12: Overview of the supported filters

Cosine Roll Off Factor Bandwidth	Root Cosine Roll Off Factor
	
Gauss (FSK) B x T	Rectangular
	

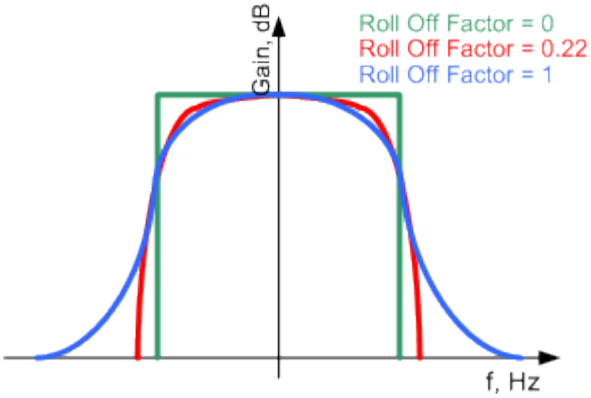
The following is a simple description of the filter parameters and the way they affect the main filter characteristics. Changing filter parameters is an effective way to ensure that the entire bandwidth of the desired signal is allowed to pass and adjust the filter form to reach spectrum mask requirements

Cut Off Frequency

The cut off frequency or corner frequency is a filter characteristic that defines the frequency at the 3 dB down point. This frequency is the bound to the transition band; here the filter characteristic changes form the passband to the stopband, where the signal is completely suppressed.

Roll Off Factor

The roll off factor is a measure for the excess bandwidth compared to the ideal bandwidth of a "brick like" filter. The roll off factor affects the steepness of the filter flanks. A "Roll Off Factor" = 0 would result in the steepest flanks theoretical possible; values near to 1 make the flanks more flat.



Passband

The passband describes the frequency span that the filter passed unchanged. The total passband of a filter is given as follow:

```
Bandwidth = (1 + "Roll Off Factor") * "Symbol Rate"
```

A.6 References

- R&S SMW200A Vector Signal Generator user manual, the latest version is available for download at R&S SMW200A product homepage.
- R&S SMBV100A Vector Signal Generator operating manual, the latest version is available for download at R&S SMBV100A product homepage.
- R&S SGT100A SGMA Vector RF Source operating manual, the latest version is available for download at R&S SGT100A product homepage.

Glossary: Terms and Abbreviations

A

AM: Amplitude modulation

Antenna: The antenna characterizes the radiation pattern of the Emitter.

ARB: Arbitrary Waveform Generator
An I/Q modulation source forming an integral part of the supported signal generators. The ARB allows the playback and output of any externally calculated modulation signal in the form of waveform file as well as the generation of multi carrier or multi segment signals from waveform files.

ASK: Amplitude-shift keying modulation

ASR: Airport Surveillance Radar

AWGN: Additive white gaussian noise

B

Barker: Special sequences (codes) that assures phase modulated signals with low autocorrelation properties.

Boresight: Antenna boresight is the direction to which an antenna shows the maximum gain.

BPSK: Binary phase shift keying modulation

BW: Bandwidth

C

C-BPSK: BPSK modulation with a constant envelope

Chirp: Signal in which the frequency varies over the time.

CW: Continuous wave signal, that is an unmodulated signal.

D

DQPSK: Differential QPSK

DUT: Device under test

E

EIRP: Equivalent isotopically radiated power

Emitter: In the context of this software, emitters emulate radar systems.

EW: Electronic warfare

F

FFT: Fast Fourier transform

FM: Frequency modulation

FSK: Frequency-shift keying modulation

G

Gain: Antenna gain is a measure of the antenna's ability to concentrate electromagnetic energy in a narrow beam.

GUI: Graphical User Interface

H

HPBW: Half-Power Beam Width

I

Inter-Pulse Modulation: IPM

Intra-Pulse Modulation: MOP

IPM: Inter-Pulse Modulation

M

MOP: Modulation on Pulse

MSW: Multi segment waveform

O

OQPSK: Offset QPSK

P

PDW: Pulse Descriptor Word
Format of the reported data

Plugin: A loadable Microsoft Windows DDL module that describes custom modulation schemes or envelope shapes.

Poly Phase: polyphase

polyphase: Code used by pulse modulation and suitable for pulse compression.

PRF: Pulse repetition frequency

PRI: Pulse repetition interval
Defines the overall time of a pulse cycle.

PRT: Pulse repetition time

Pulse sequence: Pulse train

Pulse train: A sequence of repetitive pulses

Pulse-to-Pulse Modulation: IPM

PW: Pulse width

Q

QPSK: Quadrature phase shift keying modulation

R

RADAR: Radio Detecting and Ranging

Repository: In the context of this software, a repository is a file-based database located on the local hard drive or on a network storage drive.

S

Scenario: In the context of this software, a scenario is the top level description of the signals to be generated.

Sequence: In the context of this software, a sequence describes how pulses are arranged to form a waveform.

Stagger PRI: Pulse train composed of two or more pulse sub-trains that use the same PRI.

V

VSG: Vector Signal Generator

List of Commands

*IDN? 270
*OPT? 271
*RST 271
ANTenna:CATalog? 276
ANTenna:COMMent 278
ANTenna:CREate 276
ANTenna:MODel:ARRay:COSN 284
ANTenna:MODel:ARRay:DISTribution 285
ANTenna:MODel:ARRay:NX 285
ANTenna:MODel:ARRay:NZ 285
ANTenna:MODel:ARRay:PEDestal 285
ANTenna:MODel:ARRay:XDIStance 285
ANTenna:MODel:ARRay:ZDIStance 285
ANTenna:MODel:BACKlobe:ATTenuation 286
ANTenna:MODel:BACKlobe:ENABle 285
ANTenna:MODel:BACKlobe:TYPE 286
ANTenna:MODel:BANDwidth 287
ANTenna:MODel:COSecant:HPBW 288
ANTenna:MODel:COSecant:T1 286
ANTenna:MODel:COSecant:T2 286
ANTenna:MODel:CUSTom:HPBW:XY 286
ANTenna:MODel:CUSTom:HPBW:YZ 286
ANTenna:MODel:CUSTom:SLRolloff 287
ANTenna:MODel:CUSTom:SLSCale 287
ANTenna:MODel:CUSTom:SLSTart 287
ANTenna:MODel:FREQuency 287
ANTenna:MODel:GAUSsian:HPBW 288
ANTenna:MODel:HORN:LX 288
ANTenna:MODel:HORN:LZ 288
ANTenna:MODel:PARabolic:DIAMeter 288
ANTenna:MODel:ROTation:X 289
ANTenna:MODel:ROTation:Z 289
ANTenna:MODel:SINC:HPBW 288
ANTenna:MODel:TYPE 289
ANTenna:MODel:USER:CLEar 281
ANTenna:MODel:USER:CSV:FORMat 289
ANTenna:MODel:USER:LOAD 289
ANTenna:NAME 277
ANTenna:REMove 278
ANTenna:SELect 277
ASSignment:EMITters:LIST? 302
ASSignment:EMITters:SELect 280
ASSignment:GENerator:CAPabilities? 302
ASSignment:GENerator:LIST? 302
ASSignment:GENerator:PATH:EMITter:ADD 279
ASSignment:GENerator:PATH:EMITter:DELete 281
ASSignment:GENerator:PATH:EMITter:LIST? 303

ASSignment:GENerator:PATH:EMITter:SELect....280
ASSignment:GENerator:PATH:LIST?....302
ASSignment:GENerator:PATH:SELect....280
ASSignment:GENerator:SELect....280
double PS_PLUGIN_EXPORTS getModulationClockHint....405
double PS_PLUGIN_EXPORTS getModulationSymbolRate....405
DSRC:CATalog?....276
DSRC:CLEar....281
DSRC:COMMent....278
DSRC:CREate....276
DSRC:ITEM:ADD....279
DSRC:ITEM:BITS....299
DSRC:ITEM:DATA....299
DSRC:ITEM:DELete....281
DSRC:ITEM:PATTern....300
DSRC:ITEM:PRBS:INIT....300
DSRC:ITEM:PRBS:INIT:VALue....300
DSRC:ITEM:PRBS:TYPE....300
DSRC:ITEM:SELect....280
DSRC:ITEM:TYPE....301
DSRC:NAME....277
DSRC:REMove....278
DSRC:SELect....277
EMITter:CATalog?....276
EMITter:COMMent....278
EMITter:CREate....276
EMITter:EIRP....304
EMITter:FREQuency....305
EMITter:MODE:ADD....279
EMITter:MODE:ANTenna....305
EMITter:MODE:ANTenna:CLEar....281
EMITter:MODE:BEAM:ADD....279
EMITter:MODE:BEAM:CLEar....281
EMITter:MODE:BEAM:COUNt?....280
EMITter:MODE:BEAM:DELete....281
EMITter:MODE:BEAM:OFFSet:AZIMuth....305
EMITter:MODE:BEAM:OFFSet:ELEVation....305
EMITter:MODE:BEAM:OFFSet:FREQuency....305
EMITter:MODE:BEAM:SELect....280
EMITter:MODE:BEAM:SEQuence....305
EMITter:MODE:BEAM:STATe....306
EMITter:MODE:CLEar....282
EMITter:MODE:COUNt?....280
EMITter:MODE:DELete....281
EMITter:MODE:NAME....277
EMITter:MODE:SCAN....306
EMITter:MODE:SCAN:CLEar....282
EMITter:MODE:SELect....280
EMITter:NAME....277
EMITter:REMove....278

EMITter:SELect....277
GENerator:CAPabilities....308
GENerator:CATalog?....276
GENerator:COMMent....278
GENerator:CREate....277
GENerator:LOCK....308
GENerator:NAME....277
GENerator:OPTion?....308
GENerator:REMove....278
GENerator:SELect....277
GENerator:TYPE....307
INSTrument:ADD....308
INSTrument:CAPabilities?....308
INSTrument:CLEar....281
INSTrument:COUNt?....280
INSTrument:DELete....281
INSTrument:FIRMware?....309
INSTrument:MAP....309
INSTrument:SELect....280
INSTrument:STATus?....309
int PS_PLUGIN_EXPORTS calculateNextIpmValue....409
int PS_PLUGIN_EXPORTS getBitsNeeded....406
int PS_PLUGIN_EXPORTS getNextMsg....402
int PS_PLUGIN_EXPORTS getNextReportVariable....408
int PS_PLUGIN_EXPORTS getVariable....403
int PS_PLUGIN_EXPORTS initPlugin....403
int PS_PLUGIN_EXPORTS isDataNeeded....405
int PS_PLUGIN_EXPORTS modulationCore....406
int PS_PLUGIN_EXPORTS setGlobalVariable....408
int PS_PLUGIN_EXPORTS setVariable....404
IPM:CATalog?....276
IPM:COMMent....278
IPM:CREate....277
IPM:EQUation....314
IPM:LIST:CLEar....282
IPM:LIST:ITEM:ADD....279
IPM:LIST:ITEM:COUNt?....280
IPM:LIST:ITEM:DELete....281
IPM:LIST:ITEM:REPetition....314
IPM:LIST:ITEM:SELect....280
IPM:LIST:ITEM:VALue....314
IPM:LIST:LOAD....314
IPM:LIST:SAVE....315
IPM:NAME....278
IPM:PLUGin:NAME....278
IPM:PLUGin:VARiable:CATalog?....329
IPM:PLUGin:VARiable:SELect....329
IPM:PLUGin:VARiable:VALue....329
IPM:RANDom:DISTribution....315
IPM:RANDom:NORMal:LIMit....315

IPM:RANDom:NORMal:MEAN 315
IPM:RANDom:NORMal:STD 316
IPM:RANDom:U:CENTer 316
IPM:RANDom:U:RANGe 316
IPM:RANDom:UNIForm:MAXimum 316
IPM:RANDom:UNIForm:MINimum 316
IPM:RANDom:UNIForm:STEP 317
IPM:REMove 278
IPM:RLISt:BURSt 317
IPM:RLISt:REUSe 317
IPM:RSTep:MAXimum 319
IPM:RSTep:MINimum 319
IPM:RSTep:PERiod 319
IPM:RSTep:STEP:MAXimum 319
IPM:RSTep:STEP:MINimum 319
IPM:SELect 277
IPM:SHAPe:BASE 317
IPM:SHAPe:COUNt 317
IPM:SHAPe:INTerpol 318
IPM:SHAPe:PERiod 318
IPM:STEP:BURSt 318
IPM:STEP:INCRement 318
IPM:STEP:STARt 319
IPM:STEP:STEPs 319
IPM:TYPE 313
IPM:UNIT 313
IPM:WAVeform:BASE 320
IPM:WAVeform:COUNt 320
IPM:WAVeform:OFFSet 320
IPM:WAVeform:PERiod 321
IPM:WAVeform:PHASe 321
IPM:WAVeform:PKPK 321
IPM:WAVeform:TYPE 320
PLOT:POLar:CUT 290
PLOT:POLar:LOG:MIN 290
PLOT:POLar:TYPE 290
PLUGin:CATalog? 276
PLUGin:COMMent 278
PLUGin:CREate 277
PLUGin:LOAD 328
PLUGin:MODule:AUTHor? 328
PLUGin:MODule:COMMent? 328
PLUGin:MODule:DATA? 328
PLUGin:MODule:NAME? 278
PLUGin:MODule:TYPE? 328
PLUGin:MODule:VERSion? 328
PLUGin:NAME 278
PLUGin:REMove 278
PLUGin:SELect 277
PULSe:CATalog? 276

PULSe:COMMent....278
PULSe:CREate....277
PULSe:CUSTom....335
PULSe:ENVelope:DATA:CLEar....282
PULSe:ENVelope:DATA:ITEM:ADD....279
PULSe:ENVelope:DATA:ITEM:COUNt?....280
PULSe:ENVelope:DATA:ITEM:DELete....281
PULSe:ENVelope:DATA:ITEM:SELect....280
PULSe:ENVelope:DATA:ITEM:VALue....335
PULSe:ENVelope:DATA:LOAD....335
PULSe:ENVelope:DATA:MULTiplier....336
PULSe:ENVelope:DATA:OFFSet....336
PULSe:ENVelope:DATA:SAVE....336
PULSe:ENVelope:DATA:UNIT....335
PULSe:ENVelope:EQUation....336
PULSe:ENVelope:MODE....337
PULSe:LEVel:DROop....337
PULSe:LEVel:OFF....337
PULSe:LEVel:ON....337
PULSe:MARKer:AUTO....324
PULSe:MARKer:FALL....324
PULSe:MARKer:GATE....324
PULSe:MARKer:RISE....324
PULSe:MARKer:WIDTh....324
PULSe:MOP:AM:FREQuency....337
PULSe:MOP:AM:MDEPth....337
PULSe:MOP:AM:TYPE....338
PULSe:MOP:AMSTep:ADD....279
PULSe:MOP:AMSTep:CLEar....282
PULSe:MOP:AMSTep:COUNt?....280
PULSe:MOP:AMSTep:DELete....281
PULSe:MOP:AMSTep:DURation....338
PULSe:MOP:AMSTep:INSert....281
PULSe:MOP:AMSTep:LEVel....338
PULSe:MOP:AMSTep:SELect....280
PULSe:MOP:ASK:INVert....339
PULSe:MOP:ASK:MDEPth....339
PULSe:MOP:ASK:SRATe....339
PULSe:MOP:BARKer:CODE....339
PULSe:MOP:BARKer:TTIMe....340
PULSe:MOP:BPSK:PHASe....340
PULSe:MOP:BPSK:SRATe....340
PULSe:MOP:BPSK:SRATe:AUTO....340
PULSe:MOP:BPSK:TTIMe....341
PULSe:MOP:BPSK:TTYPe....341
PULSe:MOP:BPSK:TYPE....341
PULSe:MOP:CHIRp:DEViation....341
PULSe:MOP:CHIRp:TYPE....341
PULSe:MOP:COMMent....278
PULSe:MOP:DATA:CODing....342

PULSe:MOP:DATA:DSRC....342
PULSe:MOP:ENABle....342
PULSe:MOP:EXCLude:ENABle....342
PULSe:MOP:EXCLude:LEVel:STARt....342
PULSe:MOP:EXCLude:LEVel:STOP....342
PULSe:MOP:EXCLude:MODE....343
PULSe:MOP:EXCLude:TIME:STARt....343
PULSe:MOP:EXCLude:TIME:STOP....343
PULSe:MOP:FILTer:BT....343
PULSe:MOP:FILTer:BWIDth....343
PULSe:MOP:FILTer:LENGth....344
PULSe:MOP:FILTer:ROLLoff....344
PULSe:MOP:FILTer:TYPE....344
PULSe:MOP:FM:DEViation....344
PULSe:MOP:FM:FREQuency....345
PULSe:MOP:FMSTep:ADD....279
PULSe:MOP:FMSTep:CLEar....282
PULSe:MOP:FMSTep:COUNt?....280
PULSe:MOP:FMSTep:DELete....281
PULSe:MOP:FMSTep:DURation....338
PULSe:MOP:FMSTep:FREQuency....338
PULSe:MOP:FMSTep:INSert....281
PULSe:MOP:FMSTep:SELect....280
PULSe:MOP:FSK:DEViation....345
PULSe:MOP:FSK:SRATe....345
PULSe:MOP:NLCHirp:EQUation....345
PULSe:MOP:NOISe:BWIDth....345
PULSe:MOP:PCHirp:ADD....279
PULSe:MOP:PCHirp:CLEar....282
PULSe:MOP:PCHirp:COEFficient....346
PULSe:MOP:PCHirp:COUNt?....280
PULSe:MOP:PCHirp:DELete....281
PULSe:MOP:PCHirp:INSert....281
PULSe:MOP:PCHirp:SELect....280
PULSe:MOP:PCHirp:TERM....346
PULSe:MOP:PLISt:ADD....279
PULSe:MOP:PLISt:CLEar....282
PULSe:MOP:PLISt:COUNt?....280
PULSe:MOP:PLISt:DELete....281
PULSe:MOP:PLISt:INSert....281
PULSe:MOP:PLISt:SELect....280
PULSe:MOP:PLISt:VALue....346
PULSe:MOP:PLUGin:NAME....278
PULSe:MOP:PLUGin:VARiable:CATalog?....329
PULSe:MOP:PLUGin:VARiable:SELect....329
PULSe:MOP:PLUGin:VARiable:VALue....329
PULSe:MOP:POLY:LENGth....346
PULSe:MOP:POLY:TYPE....346
PULSe:MOP:QPSK:SRATe....347
PULSe:MOP:QPSK:TYPE....347

PULSe:MOP:TYPE....347
PULSe:NAME....278
PULSe:OVERshoot....347
PULSe:OVERshoot:DECay....347
PULSe:PREView:MODE....349
PULSe:PREView:MOP....350
PULSe:REMove....278
PULSe:RIPPle....348
PULSe:RIPPle:FREQuency....348
PULSe:SELect....277
PULSe:SETTings....349
PULSe:TIME:FALL....348
PULSe:TIME:REFerence....348
PULSe:TIME:RISE....348
PULSe:TIME:WIDTh....349
PULSe:TYPE:FALL....349
PULSe:TYPE:RISE....349
REPManager:CATalog?....353
REPManager:DISCard....354
REPManager:EXPort....355
REPManager:LOAD....354
REPManager:PATH:ADD....355
REPManager:PATH:DELete....356
REPManager:PATH:LIST?....356
REPository:ACCess?....351
REPository:AUTHor....352
REPository:CATalog?....276
REPository:COMMent....278
REPository:COMPlexity....352
REPository:CREate....277
REPository:DATE....352
REPository:FILename?....352
REPository:PATH?....353
REPository:REMove....279
REPository:SAVE....353
REPository:SECurity....353
REPository:SELect....277
REPository:VERSion....353
SCAN:CATalog?....276
SCAN:CIRCular:NELevation....297
SCAN:CIRCular:NODDing....297
SCAN:CIRCular:NRATe....298
SCAN:CIRCular:PALMer....298
SCAN:CIRCular:PRATe....298
SCAN:CIRCular:PSQuint....298
SCAN:CIRCular:ROTation....292
SCAN:CIRCular:RPM....293
SCAN:COMMent....278
SCAN:CONical:RATE....293
SCAN:CONical:ROTation....292

SCAN:CONical:SQUint....294
SCAN:CREate....277
SCAN:HELical:ELEVation:STEP....293
SCAN:HELical:RETRace....295
SCAN:HELical:ROTation....292
SCAN:HELical:RPM....293
SCAN:HELical:TURNs....293
SCAN:LSW:DIRection....294
SCAN:LSW:DWELl....294
SCAN:LSW:LOBes....294
SCAN:LSW:ROTation....292
SCAN:LSW:SQUint....294
SCAN:NAME....278
SCAN:RASTer:BARS....294
SCAN:RASTer:BARWidth....295
SCAN:RASTer:DIRection....296
SCAN:RASTer:FLYBack....295
SCAN:RASTer:PALMer....298
SCAN:RASTer:PRATe....298
SCAN:RASTer:PSQuint....298
SCAN:RASTer:RATE....293
SCAN:RASTer:RETRace....295
SCAN:RASTer:UNIDirection....295
SCAN:RASTer:WIDTh....296
SCAN:REMove....279
SCAN:SECTor:FLYBack....295
SCAN:SECTor:NELevation....297
SCAN:SECTor:NODDing....297
SCAN:SECTor:NRATe....298
SCAN:SECTor:PALMer....298
SCAN:SECTor:PRATe....298
SCAN:SECTor:PSQuint....298
SCAN:SECTor:RATE....293
SCAN:SECTor:UNIDirection....295
SCAN:SECTor:WIDTh....296
SCAN:SELect....277
SCAN:SPIRal:RETRace....295
SCAN:SPIRal:ROTation....292
SCAN:SPIRal:ROUNds....296
SCAN:SPIRal:RTIMe....296
SCAN:SPIRal:STEP....297
SCAN:TYPE....297
SCENario:CACHe:VOLatile:CLEar....362
SCENario:CACHe:VOLatile:VALid?....362
SCENario:CATalog?....276
SCENario:CEMit:ADD....279
SCENario:CEMit:ALIas....363
SCENario:CEMit:CURRent....363
SCENario:CEMit:DELete....281
SCENario:CEMit:DIRection:BEARing....364

SCENario:CEMit:DIRection:ELEVation 364
SCENario:CEMit:EMITter 364
SCENario:CEMit:EMITter:MODE 365
SCENario:CEMit:EMITter:MODE:BEAM 365
SCENario:CEMit:SELect 280
SCENario:COMMent 278
SCENario:CREate 277
SCENario:CSEQuence 368
SCENario:CSEQuence:ADD 279
SCENario:CSEQuence:ALIas 363
SCENario:CSEQuence:CLEar 282
SCENario:CSEQuence:CURRent 363
SCENario:CSEQuence:DELete 281
SCENario:CSEQuence:SELect 280
SCENario:CSEQuence:VARiable 368
SCENario:EMITter 364
SCENario:EMITter:CLEar 282
SCENario:EMITter:DIRection:PHI 364
SCENario:EMITter:DIRection:THETa 364
SCENario:EMITter:MODE 365
SCENario:EMITter:MODE:BEAM 365
SCENario:GENerator 362
SCENario:GENerator:PATH 362
SCENario:LOCalized:ADD 279
SCENario:LOCalized:ALIas 363
SCENario:LOCalized:CURRent 363
SCENario:LOCalized:DELete 281
SCENario:LOCalized:DIRection:BEARing 364
SCENario:LOCalized:DIRection:ELEVation 364
SCENario:LOCalized:DIRection:TRACk 366
SCENario:LOCalized:DISTance? 365
SCENario:LOCalized:EMITter 364
SCENario:LOCalized:EMITter:MODE 365
SCENario:LOCalized:EMITter:MODE:BEAM 365
SCENario:LOCalized:LOCation:ALTitude 366
SCENario:LOCalized:LOCation:EAST 366
SCENario:LOCalized:LOCation:NORTh 366
SCENario:LOCalized:RECeiver:ALTitude 366
SCENario:LOCalized:RECeiver:ANTenna 366
SCENario:LOCalized:RECeiver:DIRection:BEARing 364
SCENario:LOCalized:RECeiver:DIRection:ELEVation 364
SCENario:LOCalized:RECeiver:GAIN 367
SCENario:LOCalized:RECeiver:SCAN 367
SCENario:LOCalized:SELect 280
SCENario:LOCalized:TYPE 367
SCENario:LOCalized:WAVeform 367
SCENario:LOCalized:WAVeform:ANTenna 366
SCENario:LOCalized:WAVeform:EIRP 367
SCENario:LOCalized:WAVeform:FREQuency 368
SCENario:LOCalized:WAVeform:SCAN 367

SCENario:NAME....278
SCENario:OUTPut:CLOCk:AUTO:BORDer....369
SCENario:OUTPut:CLOCk:AUTO:OVERsampling....369
SCENario:OUTPut:CLOCk:MODE....368
SCENario:OUTPut:CLOCk:USER....369
SCENario:OUTPut:DURation:MODE....369
SCENario:OUTPut:DURation:TIME....370
SCENario:OUTPut:FORMat....370
SCENario:OUTPut:FREQuency....370
SCENario:OUTPut:LEVel....370
SCENario:OUTPut:LOOP:COMBine....370
SCENario:OUTPut:MARKer:ENABle....325
SCENario:OUTPut:MARKer:SCENario:DURation....325
SCENario:OUTPut:MARKer:SCENario:ENABle....325
SCENario:OUTPut:PATH....371
SCENario:OUTPut:REPository:ENABle....371
SCENario:OUTPut:RESet:ENABle....371
SCENario:OUTPut:RF:ENABle....371
SCENario:OUTPut:RUNMode....371
SCENario:OUTPut:TARGet....372
SCENario:OUTPut:THReshold....372
SCENario:PDW:ENABle....329
SCENario:PDW:PATH....329
SCENario:PDW:PLUGin:NAME....330
SCENario:PDW:PLUGin:VARiable:CATalog?....329
SCENario:PDW:PLUGin:VARiable:SELect....329
SCENario:PDW:PLUGin:VARiable:VALue....329
SCENario:PDW:TEMPlate....330
SCENario:PDW:TYPE....330
SCENario:REMove....279
SCENario:SELect....277
SCENario:SEQuence....363
SCENario:SEQuence:CLEar....363
SCENario:STARt....361
SCENario:STATe?....361
SCENario:STOP....361
SCENario:TYPE....361
SCENario:VOLatile:VIEW....372
SCENario:VOLatile:VIEW:XMODe....386
SCENario:VOLatile:VIEW:YMODe....386
SCENario:VOLatile:VIEW:ZOOM:POINt....386
SCENario:VOLatile:VIEW:ZOOM:RANGe....387
SEQuence:CATalog?....276
SEQuence:COMMent....278
SEQuence:CREate....277
SEQuence:ITEM:ADD....279
SEQuence:ITEM:CLEar....282
SEQuence:ITEM:COUNt?....280
SEQuence:ITEM:DELete....281
SEQuence:ITEM:FILLer:MODE....375

SEQuence:ITEM:FILLer:SIGNal....375
SEQuence:ITEM:FILLer:TIME....375
SEQuence:ITEM:FILLer:TIME:EQUation....375
SEQuence:ITEM:FILLer:TIME:FIXed....376
SEQuence:ITEM:FREQuency:OFFSet....376
SEQuence:ITEM:INDent....376
SEQuence:ITEM:IPM:A....321
SEQuence:ITEM:IPM:ADD....279
SEQuence:ITEM:IPM:B....321
SEQuence:ITEM:IPM:COUNt?....280
SEQuence:ITEM:IPM:DELete....281
SEQuence:ITEM:IPM:MODE....322
SEQuence:ITEM:IPM:RANDom:RESet....322
SEQuence:ITEM:IPM:RESTart....322
SEQuence:ITEM:IPM:SELect....280
SEQuence:ITEM:IPM:SOURce....322
SEQuence:ITEM:IPM:TARGet:PARameter....313
SEQuence:ITEM:IPM:TARGet:TYPE....313
SEQuence:ITEM:IPM:TARGet:VARiable....314
SEQuence:ITEM:LEVel:OFFSet....376
SEQuence:ITEM:LOOP:COUNt:FIXed....377
SEQuence:ITEM:LOOP:COUNt:MAXimum....377
SEQuence:ITEM:LOOP:COUNt:MINimum....377
SEQuence:ITEM:LOOP:COUNt:STEP....377
SEQuence:ITEM:LOOP:TYPE....377
SEQuence:ITEM:LOOP:VARiable....378
SEQuence:ITEM:MARKer:ALL....324
SEQuence:ITEM:MARKer:FIRSt....324
SEQuence:ITEM:MARKer:LAST....324
SEQuence:ITEM:OVL:VARiable....378
SEQuence:ITEM:OVL:WTIMe....378
SEQuence:ITEM:PDELay....378
SEQuence:ITEM:PHASe:OFFSet....378
SEQuence:ITEM:PRI....379
SEQuence:ITEM:PULSe....379
SEQuence:ITEM:REP:COUNt:DURation....379
SEQuence:ITEM:REP:COUNt:FIXed....380
SEQuence:ITEM:REP:COUNt:MAXimum....380
SEQuence:ITEM:REP:COUNt:MINimum....380
SEQuence:ITEM:REP:COUNt:ROUNding....380
SEQuence:ITEM:REP:COUNt:STEP....380
SEQuence:ITEM:REP:TYPE....381
SEQuence:ITEM:REP:VARiable....381
SEQuence:ITEM:SELect....280
SEQuence:ITEM:TYPE....381
SEQuence:ITEM:WAVeform....379
SEQuence:NAME....278
SEQuence:REMove....279
SEQuence:SELect....277
SEQuence:TYPE....375

STATus:OPERation:CONDition?....382
SYSTem:ERRor:ALL?....382
SYSTem:ERRor:COUNt?....383
SYSTem:ERRor?....382
SYSTem:PROGress?....383
void PS_PLUGIN_EXPORTS generateDirac....407
void PS_PLUGIN_EXPORTS getAuthor....402
void PS_PLUGIN_EXPORTS getComment....401
void PS_PLUGIN_EXPORTS getError....402
void PS_PLUGIN_EXPORTS getName....402
void PS_PLUGIN_EXPORTS getType....401
void PS_PLUGIN_EXPORTS getVersion....401
void PS_PLUGIN_EXPORTS restart....409
void PS_PLUGIN_EXPORTS setBits....406
void PS_PLUGIN_EXPORTS setModulationParameter....405
void PS_PLUGIN_EXPORTS setRndFnPtr....409
void PS_PLUGIN_EXPORTS shutdownPlugin....403
WAVeform:CATalog?....276
WAVeform:COMMent....278
WAVeform:CREate....277
WAVeform:IQ:CLEar....385
WAVeform:LEVel:REFerence....385
WAVeform:NAME....278
WAVeform:NOISe:BWIDth....384
WAVeform:REMove....279
WAVeform:SELect....277
WAVeform:SIGCont....385
WAVeform:TYPE....384
WAVeform:VIEW:XMODe....386
WAVeform:VIEW:YMODe....386
WAVeform:VIEW:ZOOM:POINt....386
WAVeform:VIEW:ZOOM:RANGe....387
WAVeform:WAVeform:CLEar....385
WAVeform:WAVeform:LOAD....385

Index

Symbols

2D map
Range ... 202
3D
Rotating ... 22
Zooming, mouse wheel ... 22

A

Angular Step
Spiral scan ... 169
Antenna
Model ... 156
Antenna pattern
Comment ... 156
Name ... 156
Antenna pattern settings ... 155
Antenna scan
Comment ... 165
Name ... 165
Type ... 165
Antenna scans settings ... 164
ARB file
Create ... 59
Load ... 59
Automation ... 266
Available connected generators ... 220
Avoiding data truncation
Rising/falling time ... 72

B

Back lobe ... 154
Base power ... 73
Beam
Azimuth offset ... 182
Elevation offset ... 182
Frequency offset ... 182
Black
Item ... 110
Blank segment ... 115
Duration ... 115
Timing ... 115
Boresight ... 154
Align to ... 157
Displacement ... 157

C

Cannot change repository
Obtain write permission ... 44
Cannot store repository
Remove write lock ... 44
Carrier frequency ... 56
Change user ... 43
Circular scan ... 165
Conical scan ... 168
Create
Frequency hoping jitter ... 143
IPM ... 143
Jitter, PRI ... 143
Create user ... 43
CW
Item ... 110
CW segment ... 115
Duration ... 115
Timing ... 115

D

Data Source
Comment ... 228
Frequency ... 229
Name ... 228
Preview ... 230
Data source guard bits
Avoiding data truncation ... 72
Data source settings ... 228
Data truncation
Add guard bits ... 72
Avoiding ... 72
Default values
Remote ... 271
Delete
All pulse sequencer wavefroms form VSG ... 58
IPM ... 143
Jitter ... 143
Delete waveforms
Automatically ... 58
Delta
Frequency ... 242
Marker ... 242
x ... 242
y ... 242
Direction
Lobe switching scan ... 170
Display options
2D map ... 202
Droop ... 66
Definition ... 73
Ratio ... 73
Dummy data
Item ... 110
Dwell time
Lobe switching scan ... 170

E

Emitter
Beam ... 182
Comment ... 181
EIRP ... 181
Frequency ... 181
Name ... 181
Operating mode ... 181
Envelope functions ... 76
Error queue query ... 382, 383
Export ARB file ... 59

F

Fall time ... 66, 71
Fast data upload
FTP ... 252
Filter parameter
Impact ... 410

Flyback time
Raster scan 167
Sector scan 166
Frequency diagram 97
FTP
Upload to the signal generator 252

G

Generator profile
ARB 221
Baseband bandwidth 221
Capabilities 221
Comment 220
Features 221
Generator type 220
Installed options 221
Memory extension 221
Name 220
Profile type 220
Pulse sequencer 221
RF settings 220
Streams 221
Guest user
Access rights 43

H

Hardware
CPU 16
Requirements 16
Virtual Memory 16
Helical scan 169

I

I/Q constellation diagram 98
I/Q diagram 97
Identification
Remote 270
Installation 14
Installing
Hardware 16
Software 16
Uninstalling old version 16
Instrument
Add 218
Delete 218
IP address 218
Refresh list 218
Scan interfaces 218
Search for 218
Signal path 57
Instruments settings 216
Inter-pulse modulation (IPM) 111
Inverse pulse
Generation 73
IPM
Create 143
IPM profile
Assign to pulse parameter or variable 143
Comment 131
Name 131
Repetition 143
Select 143
Transformation 143
Type 131
Unit 131
View 131
IPM shape
Equation 136
Interpolated shape 135
List mode 133
Plugin 139
Pseudo random list 136
Random 138
Staircase 132
Steps 132
Waveform 134
Item
Blank 110
CW 110
Dummy data 110
Filler 110
Inter-pulse modulation (IPM) 111
Overly 110
Repetition 110
Sequencing 110

J

Jitter
Add 143
Create 143

L

Lobe switching scan 170
Lobes
Lobe switching scan 170
Lock symbol
Meaning 44
Locked repository
Unlock 389
Long waveform
Reducing size 248

M

Main lobe 154
Marker
Pulse 236
Sequence 236
Marker traces
Enable 242
MOP
Enable 80
Restrict area 80
MOP Type 82

N

Nodding
Circular scan 165
Sector scan 166

O

OFF time 66
ON time 66, 71
Options
Identification (remote) 271
Overlay
Item 110

Overshoot
Definition ... 74
Ratio ... 74

P

Palmer scan
Circular scan ... 165
Raster scan ... 167
Sector scan ... 166
Parameters
Base Power ... 73
Description ... 66
Droop ... 73
Fall time ... 71
IEEE 181 Standard ... 66
ON Time ... 71
Overshoot ... 74
Pulse Width ... 71
Ripple ... 75
Rise time ... 71
Top Power ... 73
PDW
Report ... 260
Phase diagram ... 97
Plugin
Variable ... 94
Power
Base ... 73
Top ... 73
Pulse
Envelope shaping, custom ... 70
Envelope shaping, standard ... 71
Fall time, show on graph ... 96
Falling edge ... 71
Falling slope ... 71
Level ... 100
MOP, modulation on pulse ... 102
Name ... 69
Period ... 66
Repetition interval ... 66
Rise time, show on graph ... 96
Rising edge ... 71
Rising slope ... 71
Shape ... 71
Shaping ... 70, 71
Show timing ... 105
Timing ... 99
Timing, visualizing ... 105
Visualizing ... 105
Width ... 66, 71
Width, show on graph ... 96
Pulse Train
Scenario ... 54

R

Radius
Conical scan ... 168
Helical scan ... 169
Raster scan ... 167
Rate
Conical scan ... 168
Helical scan ... 169
Red lock
Repository tree ... 389
Ref. level
Pulse envelope ... 56
Reference level
Measure ... 244
Remote control
Programming examples ... 270
SCPI ... 266
Remove
IPM ... 143
Jitter ... 143
Remove wavefroms ... 58
Repetition
Item ... 110
Report
Plugin ... 260
Template ... 260
Reporting
Custom template ... 263
Enable ... 262
PDW ... 262
Plugin ... 263
Storage location ... 262
Template ... 262
Template, custom ... 263
Template, edit ... 262
Type ... 262
Repository
Author ... 42
Comment ... 43
Complexity level ... 42
Copy to ... 47
Crash ... 389
Delete ... 47
Discard ... 47
Export ... 47
List of ... 47
Load ... 47
Location ... 44
Name ... 42
Obtain write permission ... 44
Open in Explorer ... 44
Path, add ... 47
Remove write lock ... 44
Search ... 47
Security level ... 42
Users ... 43
Reset values
Remote ... 271
Restart
Connected generator ... 56
Retrace Time
Spiral scan ... 169
Ripple ... 75
Definition ... 75
Rise time ... 66, 71
Round Time
Spiral scan ... 169
Rounds
Spiral scan ... 169

S

Sanitize ... 58
Scan rate
Circular scan ... 165
Raster scan ... 167
Sector scan ... 166

Scenario
Busy 57
Comment 55
Create ARB file 59
Frequency 56
Name 55
Play mode 57
Ref. level 56
Reset connected generator 56
Run 57
Settings 54
Start 57
Stop 57
Type 54, 55
Sector scan 166
Sector width
Raster scan 167
Sector scan 166
Security level
Repository 42
Sequence
Comment 109
Name 109
Select item 109
Type 109
Sequence settings 108
Show timing 96
Side lobe 154
Signal generation
Start 57
Socket communication 266
Software
Requirements 16
Spectrum 98
Spiral scan 169
Squint angle
Lobe switching scan 170
Start delay
1 ms 258
Storing
Reports 262

T

Time delay
1 ms 258
Time synchronization
Filler 115
Top power 73

U

Unidirectional
Raster scan 167
Sector scan 166
Upload
Speed up 252
User account 52
User data source
Rules 227
User manual 10
User name
Custom 52
Set default user 52
User roles 43

V

Varying
Pulse parameters 143
View
IPM profile 131

W

Waveform
AWGN Bandwidth 194
Clear 194
Comment 194
Import 194
Name 193
Reference level 194
Signal content 194
Type 194
View 194
Visualize 194
Waveform size
Reducing 248
Write lock
Remove 389

Z

Zooming
With mouse in 3D graphs 22